# 4 Modelado de dominio

## Diccionario de datos

### Entidades

|  |  |  |
| --- | --- | --- |
| CLIENTES | | |
| Particular: bool | En el caso de no ser un particular, se considera que es una empresa. Da la información de cómo interpretar el resto de los campos (DNI/CIF o nombre). | not null |
| DNI/CIF: char [10] | En caso de que sea un particular se introducirá el DNI, si es una empresa se introduce el CIF correspondiente. | PK-not null |
| Nombre |  | Not null |
| Teléfono | Teléfono de contacto proporcionado por el cliente, puede ser un móvil o un fijo. | Not null |
| Dirección | Comunidad, población, código postal, vía, número y puerta | Not null |
| e-mail |  | Not null |

Clientes: trataremos de almacenar la máxima información posible de ellos, dejando como no obligatorio el email. Se pedirá un número de teléfono de contacto para la comunicación de incidencias.

|  |  |  |
| --- | --- | --- |
| PROVEEDORES | | |
| CIF: char [10] | Los proveedores son empresas que se dedican al suministro de las piezas, por lo que se les pedirá el CIF. | PK-not null |
| Nombre |  | Not null |
| Teléfono |  | Not null |
| Dirección | Comunidad, población, código postal, vía, número y puerta | Not null |
| e-mail |  | Not null |

Proveedores: al igual que con los clientes trataremos de almacenar la máxima información posible. Cada proveedor podrá vender una o varias piezas

|  |  |  |
| --- | --- | --- |
| PIEZAS | | |
| Código | Código de referencia único para cada pieza | PK-not null |
| ID | Identificador único de cada pieza | PK-not null |
| Precio\_venta | Precio que se cobra a los clientes cuando se utiliza la pieza en un trabajo | Not null |
| Precio\_compra | Precio que se paga al proveedor cuando la pieza es pedida | Not null |
| Cantidad\_almacen | Cantidad de ese tipo de piezas disponibles en el almacén, para ayudar a la elaboración de los pedidos de piezas | Not null |

Piezas: almacenamos de cada pieza su código (conjunto de letras y número que utilizamos como identificador del tipo de pieza), el precio y la cantidad de piezas que disponemos en el almacén (para tener el control del stock que poseemos y realizar los pedidos correspondientes). Piezas tendrá una relación de herencia (tipo\_pieza) que diferenciará entre piezas\_básicas y piezas\_especiales, dicha relación es completa y exclusiva de modo que en la implementación de la base de datos crearemos dos tablas distintas, una para cada tipo de pieza. Esto se realiza por dos razones, cada tipo de pieza tiene atributos distintos por lo que de estar todas en una misma tabla habría en ella un gran desaprovechamiento de datos debido al exceso de nulls. La segunda razón es que a la hora de realizar pedidos no se pueden mezclar los tipos de pieza, la búsqueda de todas las piezas de un tipo concreto será más rápida si cada tipo está en su propia tabla.

|  |  |  |
| --- | --- | --- |
| PIEZAS\_BÁSICAS | | |
| Cantidad\_pedido | Cantidad que es necesaria pedir | not null |
| Stock\_mínimo | Cantidad por debajo de la cual se debe realizar el pedido automático | Not null |

|  |  |  |
| --- | --- | --- |
| PIEZAS\_ESPECIALES | | |
| Fecha\_uso | Fecha para la cual es necesaria la pieza |  |
| Cantidad\_necesaria | Cantidad que es necesario pedir |  |

Piezas especiales: La información de fecha de uso y de cantidad necesaria se guarda información adicional para ayudar al **responsable del almacén** a realizar los pedidos manuales. Si sabe cuantas y para cuando se necesitan las piezas podrá pedirlas manualmente de forma más sencilla.

|  |  |  |
| --- | --- | --- |
| TRABAJADORES | | |
| DNI: char [10] |  | PK-not null |
| Nombre |  | Not null |
| Teléfono |  | Not null |
| Dirección |  | Not null |
| e-mail |  | Not null |
| Estado | Informa sobre si el trabajador está de vacaciones, enfermo u otra situación en la que no se le deban asignar peticiones |  |

Trabajadores: almacenamos la información básica de cada trabajador, para diferenciar si es un coordinador técnico, un ayudante... De trabajadores tendremos una relación de herencia completa y exclusiva (**cargo**) que nos ayuda a diferenciar los distintos cargos de trabajador que tenemos en la empresa (**técnicos informáticos**, **coordinadores técnicos**, **ayudantes técnicos**, **responsables de almacén** y **ayudantes de almacén**). A efectos de nuestro sistema solo nos será relevante el **técnico informático**, los demás solo nos interesarán por mantener un listado de los trabajadores.

A efectos de la base de datos existirán cinco tablas, una por cada tipo de trabajador pues de este modo podemos controlar de forma mejor el acceso a los datos y la visualización de cada tipo de trabajador ya que por lo general no se va a querer acceder a la de todos lo trabajadores a la vez si no solo a la de un tipo concreto.

|  |  |  |
| --- | --- | --- |
| OFICINA | | |
| Dirección | Guardamos la dirección de la oficina, la cual utilizamos como única porque dos oficinas no estarán en la misma | PK-not null |
| Tipo oficina | La oficina puede ser pequeña, mediana o grande en función del número de trabajadores que posea. | Not null |

Oficina: Como hemos explicado antes, tenemos distintos grados de importancia en las oficinas, principalmente la importancia de una oficina la diferencia del resto por tener más o menos **trabajadores** en ella.

|  |  |  |
| --- | --- | --- |
| PARTE TRABAJO | | |
| Fecha\_inicio | Se almacenará el día, mes, año y hora en la que el técnico comienza su trabajo como timestamp | PK-not null |
| Precio hora | Dentro del parte de trabajo almacenamos el precio de la hora de trabajo, ya que esta podrá variar con el tiempo | Not null |
| Fecha\_fin | Se almacenará el día, mes, año y hora en la que el técnico finaliza su trabajo | Not null |

Parte de trabajo: del parte de trabajo almacenaremos un código que nos identifique el parte como único (para evitar conflictos en el caso de que varios partes se crearan simultáneamente), las horas totales empleadas (calculadas a partir de fecha\_hora\_inicio y fecha\_hora\_fin) y el precio de la hora (útil para cuando se realice la factura asociada al parte de trabajo)

|  |  |  |
| --- | --- | --- |
| PETICIÓN DE TRABAJO | | |
| Numero | Numero que identifica de forma única al parte de trabjo | PK-not null |
| Fecha | Se registra el momento concreto en el que el cliente solicitó la petición de trabajo, de forma que la identifica a modo de timestamp | Not null |
| Descripción | Se aporta una breve descripción de lo que el cliente solicita | Not null |
| Prioridad | Es asignada por el coordinador técnico y es útil a la hora de hacer la asignación del técnico responsable de un trabajo | Not null |
| Estado | Indica el estado del pedido (los posibles estados están explicados en el diagrama de estados correspondiente) | Not null |

|  |  |  |
| --- | --- | --- |
| PRESUPUESTO | | |
| Numero | Numero que identifica de forma única al presupuesto | PK-not null |
| Fecha emisión | Se almacena la fecha en la que se emite un presupuesto como timestamp | PK- not null |
| Importe\_estimado | Se estima el importe del presupuesto a partir del material que sea necesario y el total de horas que se estime que se necesitara para llevar a cabo el trabajo |  |
| Total\_horas\_estimado | Se estima el total de horas que supondrá realizar el trabajo, para calcular el importe, pero también para ayudar a la labor de asignar una prioridad al trabajo |  |

|  |  |  |
| --- | --- | --- |
| PEDIDO | | |
| Numero | Numero que identifica de forma única al pedido | PK-not null |
| fecha | Fecha en la que se realiza el pedido al proveedor | not null |
| Estado\_pedido | Se indica el estado en el que se encuentra el pedido (los posibles estados se detallan en el correspondiente diagrama de estados) | not null |
| Tipo\_pedido | Es un booleanos que al ser true indica que el pedido contiene piezas | not null |

Pedido: existe una relación de herencia (**tipo\_pedido**) que es completa y exclusiva de modo que en la base de datos se crearán una única tabla para almacenar los pedidos. Cada pedido tipo se diferenciará solo mediante la variable tipo de pedido que indica qué tipo de piezas contendrá dicho pedido. Se diferencia por tanto entre **pedido básico** y **pedido especial**.

|  |  |  |
| --- | --- | --- |
| FACTURA | | |
| numero\_factura | Identifica a la factura de forma única | PK-not null |
| fecha\_factura | Fecha en la que se emitió la factura al cliente, puede estar creada pero todavía no emitida así que la fecha podría ser nula antes de que la factura se enviara al cliente | nullable |
| Importe\_factura | Nos indicará el precio final que el cliente deberá abonar, aplicando los descuentos (en caso de tenerlos) y el IVA | Not null |

### Relaciones

|  |  |  |
| --- | --- | --- |
| SUMINISTRA | | |
| Cardinalidad | Entidad | Características |
| 1:N | Proveedor | Un proveedor vende 1 o n piezas |
| 1:1 | Piezas | Cada pieza tiene asociado un proveedor |

Suministra: es una relación entre proveedores y piezas. Un proveedor puede suministrar una o varias piezas, pero cada pieza es suministrada por un proveedor.

|  |  |  |
| --- | --- | --- |
| TRABAJA EN | | |
| Cardinalidad | Entidad | Características |
| N:1 | Trabajador | Varios trabajadores pueden trabajar en una oficina |
| 1:N | Oficina | En una oficina trabajan varios trabajadores |

Trabaja en: Los trabajadores trabajan en una oficina, que puede ser pequeña, mediana o grande en función del número de trabajadores que tiene.

|  |  |  |
| --- | --- | --- |
| SE ASIGNA | | |
| Cardinalidad | Entidad | Características |
| 0:N | Técnico informático | Un técnico informático puede no tener ninguna petición de trabajo asignada en un momento dado o tener múltiples |
| 0:1 | Petición de trabajo | Una petición de trabajo puede no tener aún asignada ningún técnico, pero en caso de tener técnico solo puede ser 1 |

Se asigna: Una petición de trabajo es asignada a un técnico informático una vez el estado está marcado como pendiente (tras que el cliente apruebe el presupuesto, en caso de solicitarlo). Justo después de esta asignación, el estado se actualiza. Un técnico puede tener asignado varias o ninguna petición de trabajo, pero una petición de trabajo puede tener como máximo un técnico asignado.

|  |  |  |
| --- | --- | --- |
| SE PLANIFICA | | |
| Cardinalidad | Entidad | Características |
| 0:1 | Petición de trabajo | Una petición de trabajo puede no tener un presupuesto (porque el cliente no lo haya solicitado) |
| 1:1 | Presupuesto | Un presupuesto siempre tiene asignado una petición de trabajo. |

|  |  |  |
| --- | --- | --- |
| SE REALIZA | | |
| Cardinalidad | Entidad | Características |
| 0:1 | Petición de trabajo | Una petición de trabajo puede no tener aún ningún parte de trabajo (porque no tenga asignado todavía técnico o porque el técnico aún no haya registrado ningún parte de trabajo) |
| 1:1 | Parte de trabajo | Un parte de trabajo siempre tiene asignada una petición de trabajo. |

Se realiza: un parte de trabajo se realiza en función de una petición de trabajo.

|  |  |  |
| --- | --- | --- |
| TIENE | | |
| Cardinalidad | Entidad | Características |
| 0:N | Piezas | Una pieza puede no aparecer en ningún presupuesto, o aparecer en varios |
| 0:N | Presupuesto | En un presupuesto puede no aparecer ninguna pieza o varias |
| Atributo | **CANTIDAD** | Indica la cantidad de un tipo de piezas que han sido pedidas en un pedido |

|  |  |  |
| --- | --- | --- |
| TRAE\_BASICO | | |
| Cardinalidad | Entidad | Características |
| 1:N | Pedido\_basico | Un pedido tiene que tener como mínimo una pieza básica |
| 0:N | Piezas\_basicas | Una pieza básica puede estar en 0 o N pedidos básicos |
| Atributo | **CANTIDAD** | Indica la cantidad de un tipo de piezas básicas que han sido pedidas en un pedido |

|  |  |  |
| --- | --- | --- |
| TRAE\_ESPECIAL | | |
| Cardinalidad | Entidad | Características |
| 1:N | Pedido\_especial | Un pedido tiene que tener como mínimo una pieza especial |
| 0:N | Piezas\_especiales | Una pieza especial puede estar en 0 o N pedidos especiales |
| Atributo | **CANTIDAD** | Indica la cantidad de un tipo de piezas especiales que han sido pedidas en un pedido |

Es necesario realizar dos relaciones para la acción de traer piezas pues un pedido solo puede crear piezas de un mismo tipo, por tanto, es necesario crear dos tipos distintos de pedido y dos relaciones distintas.

|  |  |  |
| --- | --- | --- |
| REALIZADA\_EN | | |
| Cardinalidad | Entidad | Características |
| 1:1 | Oficina | Las peticiones se asignan a una y solo una oficina |
| 0:N | Petición de trabajo | Una petición puede estar en 0 o N pedidos especiales |

|  |  |  |
| --- | --- | --- |
| ALMACENADA\_EN | | |
| Cardinalidad | Entidad | Características |
| 1:1 | Oficina | Las piezas se asignan a una y solo una oficina |
| 0:N | Pieza | Una petición puede estar en 0 o N pedidos especiales |

|  |  |  |
| --- | --- | --- |
| PEDIDO\_EN | | |
| Cardinalidad | Entidad | Características |
| 1:1 | Oficina | Los pedido se asignan a una y solo una oficina |
| 0:N | Pedido | Una petición puede estar en 0 o N pedidos especiales |

|  |  |  |
| --- | --- | --- |
| CONSUME | | |
| Cardinalidad | Entidad | Características |
| 0:N | Pieza | Una pieza puede estar en 0 o N partes de trabajo |
| 0:N | Parte de trabajo | Una parte de trabajo puede tener 0 o N piezas |
| Atributo | **CANTIDAD** | Indica la cantidad de un tipo de piezas que han sido utilizadas |

|  |  |  |
| --- | --- | --- |
| FACTURADO | | |
| Cardinalidad | Entidad | Características |
| 1:1 | Petición de trabajo | Una factura tendrá una y solo una petición de trabajo |
| 1:0 | Factura | Una petición de trabajo tendrá 1 o ninguna factura |

## Formato de datos no perteneciente al modelo entidad relación

Para funcionar la empresa hace uso de los datos mencionados anteriormente. Dichos datos son almacenados en la base de datos de la empresa para para poder acceder a ellos. De forma interna la aplicación utilizará el modelo relacional en el que los datos se guarden, no obstante, para proporcionar información a los usuarios del sistema los datos son formateados con otras estructuras más adecuadas al uso humano.

Dichas estructuras son una vista de los datos almacenados de forma que estos sean más accesibles, no obstante, no forman parte ni del modelo entidad relación ni del modelo relacional de la base de datos.

|  |  |
| --- | --- |
| Informe de beneficios | |
| Ingresos | Dinero que ha ingresado la empresa en la semana |
| Gasto en piezas |  |
| Pago a trabajadores |  |
| Trabajadores contratados | Total de trabajadores a los que paga la empresa |
| Sueldo medio de los técnicos | Sueldo medio que se paga a los técnicos |
| Gasto en I+D+I | Gasto en investigación que realiza la empresa |
| Gastos adicionales |  |
| Beneficios | Dinero neto que se ha ganado |
| Beneficio medio | Se utiliza para saber si el beneficio de la semana está por encima de lo normal o no |

Se proporciona un **informe de beneficios** nuevo al **coordinador técnico** cada semana.

|  |  |
| --- | --- |
| Resumen de trabajo | |
| Trabajos nuevos |  |
| Media de trabajos nuevos | Sirve para comparar con los trabajos nuevos en la semana para saber si son más o menos que la media. |
| Trabajos completados |  |
| Media de trabajos completados | Sirve para comparar con los trabajos completados en la semana para saber si son más o menos que la media. |
| Trabajos pendientes | Dice los trabajos que están sin asignar ningún técnico o que están a la espera de presupuesto. |
| Tasa de acumulación | Informa sobre si quedan más o menos trabajos pendientes que en la semana anterior. |
| Personal extra actual | Informa sobre la cantidad de personal adicional al base que hay contratado. |

Se proporciona un **resumen de trabajo** nuevo al **coordinador técnico** cada semana.

## Modelo Entidad-Relación

A picture containing screenshot

Description automatically generated

## Evolución de estado de los partes de trabajo.

