

velneo®



**Aprende a programar software de
gestión en Velneo**

Introducción

Realiza este curso básico dividido en mini-vídeos y haz un recorrido por los elementos fundamentales de la plataforma a la vez que desarrollas una aplicación de facturación.

Con este curso más el servidor de pruebas que te damos, en 1 semana habrás desarrollado tu primera aplicación empresarial.

Paso 1: [Crea tu Velneo vServer de pruebas.](#)

Paso 2: Instala la solución con el ejemplo que vas a desarrollar. La puedes instalar de 2 formas diferentes:

A) Desde vDevelop utilizando la opción de menú Soluciones/Asistente de primeros pasos.

B) [Descarga](#) el fichero con la instalación de la aplicación desarrollada durante el curso con los datos de ejemplo.

Una vez realizada la descarga de este fichero utiliza el componente vAdmin y la opción Archivo/Instalar solución y sigue los pasos del asistente hasta completar la instalación de la solución desarrollada durante el curso con los datos de ejemplo.

Paso 3: Sigue el temario del curso que te proponemos a continuación.

Temario: aprende a programar software de gestión

Curso de iniciación: Mi primera aplicación

Curso básico en el que darás un recorrido a los elementos fundamentales de la plataforma a la vez que desarrollas una aplicación de gestión. Tras su realización, estarás iniciado en Velneo y conocerás los elementos que usarás en el 90% de tu programación.

1. Planteamiento

En este vídeo se repasan las historias de usuario que el cliente final nos plantea para el desarrollo de la aplicación de gestión.

2. Análisis

A partir de las historias de usuario o requisitos que el cliente nos plantea analizamos las mismas convirtiendo cada una de ellas en objetos o reglas de negocio a programar.

3. vGestión. La aplicación que vamos a desarrollar

Te muestro en este vídeo la aplicación que vamos a desarrollar durante el curso para que te hagas una idea aproximada de todo lo que vamos a aprender a desarrollar con Velneo en tiempo récord.

4. Documentación Velneo

Te desglosamos los diferentes canales formativos y de documentación de la plataforma Velneo para que puedas obtener el que deseas en cada momento. Documentación, manuales, vídeo tutoriales, foto, centro de soporte, vTutor, etc.

5. Servidor de desarrollo



Para comenzar a desarrollar nuestra aplicación necesitamos un servidor de edición. Te enseñamos como registrarte en la web de Velneo para crear tu servidor Velneo Cloud y configurarlo antes de empezar con el desarrollo de la aplicación.

6. Solución y proyectos

Antes de comenzar a desarrollar nuestra primera aplicación vamos a repasar esquemáticamente como es una aplicación desarrollada con Velneo, conoceremos qué es una solución, los proyectos y las instancias para desplegar el código desarrollado en una aplicación para múltiples clientes finales.

7. Creando la solución y los proyectos

Comenzamos el desarrollo de la aplicación. Configuramos nuestro editor rápidamente, nos conectamos al servidor de desarrollo en el cloud de Velneo y creamos la solución y los proyectos de aplicación que hereda el proyecto de datos y además asignamos el alias a los proyectos.

8. Base de datos. Organización en carpetas

Una de las buenas prácticas más importantes en el desarrollo de aplicaciones es la organización de los objetos que componen la aplicación. Te mostramos utilizando como ejemplo las buenas prácticas aplicadas en Velneo vERP, un componente de código fuente abierto, altamente personalizable los criterios que aplicamos en Velneo para el desarrollo de nuestras aplicaciones.

9. Base de datos. Esquemas

Crearemos nuestra primera carpeta y el primer objeto de nuestro proyecto de datos. Los esquemas nos van a permitir definir las tablas y sus

relaciones de forma visual.

10. Base de datos. Dibujos

Es conveniente crear objetos dibujos con los iconos de cada una de las tablas de nuestra aplicación para poder aplicarlos en los esquemas, formularios y opciones de menú. Veremos cómo crear los objetos dibujo y cómo importar los iconos para cada una de las tablas.

11. Base de datos. Tabla de clientes

Creamos la primera tabla de nuestra aplicación, la tabla de clientes que será de tipo maestra normal con clave numérica, como lo son la mayoría de las tablas de una aplicación y una vez creada e incorporada al esquema le asignamos el icono correspondiente.

12. Base de datos. Tabla de proveedores

Creamos la segunda tabla de nuestra aplicación, la tabla de proveedores, y lo hacemos copiando la tabla de clientes y cambiando el identificador y nombre plural y singular en la copia creada, para posteriormente añadir la tabla de proveedores y su icono al esquema. Además, hemos comenzado a utilizar abreviaturas para los nombres de las tablas y objetos en general.

13. Base de datos. Tabla de artículos

Creamos la tabla de artículos desde el editor de esquema, del tipo maestro normal con clave numérica y le asignamos el icono correspondiente en el nodo del esquema.

14. Base de datos. Tabla de familias

Creamos la tabla de familias que es de tipo maestro con clave arbolada. El código es alfanumérico y permite crear múltiples niveles en función de las

partes comunes del código. Como siempre una vez creada la tabla se aplica una abreviatura a su identificador y se le asigna el icono correspondiente en el esquema.

15. Base de datos. Relaciones entre tablas

Crearemos nuestra primera relación entre las tablas de familias y artículos, usando el criterio de que un artículo pertenece a una familia y que una familia puede tener muchos artículos. La relación se crea usando el botón crear enlace plural del editor de esquemas. Esa relación crea un campo puntero a familia en el artículo, un índice de tipo acepta repetidas por ese campo y en la familia se crea una enlace plural. Esta relación permite navegador manualmente o de forma programada desde un artículo a su familia y desde una familia a sus artículos, y además cuando estemos en un artículo tendremos toda la información de su familia sin necesidad de programar.

16. Base de datos. Añadir campos a una tabla

Añadiremos campos para dejar completas las tablas de clientes, proveedores y artículos. Repasamos de forma rápida los tipos de campos alfabéticos y los de tipo objeto.

17. Base de datos. tabla de factura de venta

Creamos la tabla de facturas de venta, cabecera de las facturas de venta con la información de la fecha, cliente y total factura, asignándole en el esquema el icono correspondiente.

18. Base de datos. Tabla de facturas de compra

Creamos la tabla de facturas de compra, cabecera de las facturas de compra con la información de la fecha, proveedor y total factura,

asignándole en el esquema el icono correspondiente.

19. Base de datos. Tabla de movimientos de almacén

Creamos la tabla de movimientos de almacén que usaremos tanto para las líneas de detalle de las facturas de venta como las de compra. Se incluyen los primeros campos operativos, entre ellos el campo fórmula numérica importe cuyo valor es el resultado del cálculo de una fórmula.

20. Base de datos. Tabla estática de tipos de movimientos

Creamos una tabla estática con los tipos de movimientos de almacén, le asignamos un icono a cada tipo de movimiento y añadimos un campo en la tabla de movimientos punteo a la tabla estática creada. Resolvemos el contenido inicial del campo tipo de movimiento en función de si es un registro de compra o de venta.

21. Base de datos. Completando la tabla de movimientos

Para completar la tabla de movimientos de almacén añadimos el contenido inicial del precio para que asuma el precio de venta o de coste del artículo, nuevos campos también añadidos a la tabla de artículos. Finalmente añadimos el campo porcentaje descuento del movimiento y modificamos la fórmula del campo importe para que se calcule como el neto de la cantidad por el precio menos el descuento redondeado a 2 decimales.

22. Base de datos. Tabla de descuentos por cliente y familia

Creamos la tabla de descuentos por cliente y familia que permite gestionar los descuentos de un cliente en múltiples familia y viceversa, los descuentos de una familia para múltiples clientes. En la tabla tendremos el campo % descuento a aplicar en las ventas.

23. Base de datos. Tabla de estadísticas por cliente, artículo y año

Creamos la tabla de estadísticas por cliente, artículo y año que almacenará el valor acumulado de las ventas de un cliente, artículo y año, así como los costes, el importe y el porcentaje de beneficio de las ventas.

24. Base de datos. Campos especiales de la tabla de movimientos

Añadimos a la tabla de movimientos los campos cliente, familia y año para poder usarlos en la resolución de los enlaces que necesitaremos para apuntar desde cada movimiento a las tablas de descuentos y estadísticas.

25. Base de datos. Apuntando a descuentos y estadísticas

Añadimos a la tabla de movimientos los campos de tipo puntero indirecto que nos permiten apuntar desde el registro del movimiento de almacén al de descuento por cliente y familia y al de estadísticas por cliente, artículo y año para poder obtener la información de esos registros e incluso poder modificar datos en dichos registros.

26. Base de datos. Refrescar datos de maestros en plurales

Para garantizar que los datos de los movimientos de almacén sean siempre coherentes con los de la cabecera de factura de venta, creamos un evento de tabla (trigger) posterior a modificación de la factura de venta que se encargará de refrescar la fecha y el cliente de la factura en los movimientos de almacén si alguno de estos datos cambia.

27. Base de datos. Actualizaciones a facturas de compra y venta

Aprenderemos a crear de forma sencilla, rápida y fiable cálculos automáticos. En este caso cuando se da de alta una línea de detalle de una factura de compra o de venta mediante una actualización estaremos automatizando el cálculo del total de la cabecera de factura de compra o venta correspondiente.

28. Base de datos. Actualizaciones condicionadas

Crearemos actualizaciones a las tablas de artículos para que se actualice el precio de coste o de venta cuando se graben líneas de detalle de facturas de compra o venta respectivamente, pero sólo cuando el precio grabado en la línea de detalle sea mayor que el que tiene como tarifa de compra o venta el artículo. De igual forma actualizaremos el descuento para ese cliente y familia cuando el descuento de la línea sea distinto de cero y sea la primera vez que se aplica para ese cliente y familia.

29. Base de datos. Actualizaciones a estadísticas

Completamos las actualizaciones necesarias para cubrir todas las historias de usuario solicitadas por el cliente a nivel de reglas de negocio en la base de datos con la actualización de los importes de venta y coste en la tabla de estadísticas. Para hacerlo de forma correcta hemos creado en la tabla de movimientos de almacén los campos precio e importe de coste de la línea de detalle de venta.

30. Base de datos. Crear registros de punteros indirectos 1ª parte

Cuando declaramos punteros de tipo indirecto real a los que queremos hacer actualizaciones debemos encargarnos de crear los registros a actualizar la primera vez que no existen, para hacerlo se suelen crear procesos o funciones como en este caso encargadas de dar el alta. Esta función es ejecutada tanto desde el evento de tabla (trigger) anterior al alta y también anterior a modificación por si cambia alguno de los valores que intervienen en el índice de clave única.

31. Base de datos. Crear registros de puntos indirectos 2ª parte

Crearemos 2 funciones para dar altas de las tablas de descuentos por cliente y familia y una segunda función para el alta de la tabla de estadística

por cliente, artículo y año. Estas funciones son ejecutadas en las altas o modificaciones de las líneas de venta.

32. Objetos visuales. Marco AUTOEXEC

Toda aplicación necesita un marco llamado AUTOEXEC para poder ser ejecutada, este marco permite configurar el icono y el título de la ventana principal de nuestra aplicación así como definir cuál será la barra de menú principal y docks donde podemos ubicar toolbars y formularios.

33. Objetos visuales. Crear instancias en la primera ejecución

La primera vez que ejecutamos nuestra aplicación se mostrará un mensaje solicitando que creamos las instancias si no existen y que nos asignemos permiso para dichas instancias en caso de que no lo tengamos. El mensaje permite lanzar el asistente para crear nuestras instancias y asignarnos los permisos. Una vez creadas las instancias la aplicación se ejecuta. Revisamos como al crearse la instancia de datos se han creado en el disco del servidor los ficheros correspondientes a las tablas declaradas en el proyecto de datos.

34. Objetos visuales. Organización en carpetas

Crearemos las carpetas para organizar los objetos visuales de la tabla de clientes. Esta organización nos sirve de ejemplo ya que la usaremos con todas las tablas.

35. Objetos visuales. Layouts

Antes de programar la interfaz de los formularios es conveniente conocer y dominar los layouts, contenedores de controles que permiten ajustar la posición y tamaño de los mismos en base a 3 tipos: horizontales, verticales y grid o rejilla. Veremos como existen layouts a nivel de formulario, también

disponen de layout los controles caja de grupo y dibujo. Existe controles específicos que se complementan con los controles de espaciadores fijo y expansible para tener un completo control sobre las posiciones de los controles en los formularios.

36. Objetos visuales. Formulario de clientes 1ª parte

Crearemos desde la galería de objetos el formulario de clientes usando el asistente de formularios en el que podremos seleccionar, la tabla, los campos, los botones y algunas configuraciones visuales a aplicar en el formulario. El resultado será un formulario que nos servirá de base para crear nuestra primera plantilla de formularios.

37. Objetos visuales. Formulario de clientes 2ª parte

Vamos a configurar el formulario para convertirlo en una plantilla reutilizable para el resto de formularios a crear en nuestra aplicación. En la primera fase ajustaremos el tamaño del formulario y el layout de botones tanto a nivel de posiciones, tamaños como de identificadores y descripción de sus controles.

38. Objetos visuales. Formulario de clientes 3ª parte

En esta parte vamos a repasar tanto el layout de detalle donde ubicamos el separador de formularios como el layout de cabecera con los datos del cliente. Es importante ser riguroso con las posiciones, tamaños y alineamientos.

39. Objetos visuales. Formulario de clientes 4ª parte

Completamos el formulario revisando el orden de tabulación, los identificadores, descripciones, icono de formulario, margen inferior de layouts y añadiendo el botón que permitirá avanzar de control con la tecla intro en lugar de usar el tabulador. Ya tenemos listo nuestro primer

formulario que nos servirá de plantilla.

40. Objetos visuales. Rejilla de clientes

Creamos el objeto rejilla (grid) de clientes que nos servirá para mostrar listas de registros de la tabla de clientes. Configuramos diferentes aspectos de la rejilla como el pie, el ancho de las columnas, sus identificadores y cálculos en el pie con el objetivo de usar esta rejilla como plantilla de futuras rejillas.

41. Objetos visuales. Búsqueda de clientes

Para preparar una opción de menú que muestre la lista de clientes por orden alfabético, una vez creada la rejilla (grid) necesitamos crear un objeto que busque todos los registros de la tabla clientes y los devuelva ordenados. Creamos una búsqueda con un sólo componente de búsqueda que usa el índice por nombre que nos devolverá la lista de clientes por orden alfabético.

42. Objetos visuales. Acción de clientes

Los objetos en Velneo tiene entrada y salida de tipo ninguno, ficha o lista. El objeto acción permite disparar objetos de forma concatenada haciendo que la salida de un objeto sea la entrada del objeto siguiente. Utilizando esta característica creamos la acción de clientes que dispara la búsqueda de clientes y la lista obtenida se la envía al segundo objeto que es la rejilla de clientes. Una vez preparada la acción se puede insertar en una menú, toolbar, etc.

43. Objetos visuales. Opción clientes en toolbar principal

Una vez creados los objetos básicos de un mantenimiento para la tabla clientes ya podemos añadir a la toolbar principal la opción Clientes que ejecutará la acción creada y nos permitirá en ejecución añadir, modificar y

eliminar clientes.

44. Objetos visuales. Localizador de clientes

Uno de los objetos que conviene crear para las tablas maestras es el localizador. Este objeto permite seleccionar un registro de la tabla maestra cuando estamos editando registros de una tabla plural. Conviene crear rejillas especiales para selección de registros que no tengan pie, ni formularios ni menú contextual. El localizador conviene que tenga declarados los índices manualmente para poder controlar que índices podrá usar el usuario final, en el ejemplo se han añadido el de trozos, nombre y palabras.

45. Objetos visuales. Toolbar de lista

En la mayoría de los objetos rejilla (grid) o alternadores se suele incluir una toolbar que permite ejecutar las acciones de alta, editar o suprimir además de otras básicas como opciones de lista o vistas alternativas. Crearemos una toolbar estándar para objetos de lista añadiendo primero los iconos necesarios así como las acciones y finalmente la toolbar que asignaremos a la rejilla de clientes.

46. Objetos visuales. Subformulario de observaciones del cliente

Un objeto habitual en los formulario de tablas maestras son los subformularios. En este caso creamos el subformulario que permite editar el campo observaciones de la tabla de clientes y lo añadimos al separador de formulario que existe en el formulario principal de clientes.

47. Objetos visuales. Proveedores

Una vez completado el trabajo de crear con mimo los objetos visuales de la tabla de Clientes crearemos en segundos los objetos visuales de la tabla de proveedores, en primer lugar porque son tablas similares pero sobre todo

porque usaremos la funcionalidad de pegar como... que nos permite copiar la carpeta de objetos visuales de clientes y al pegarla como y resolver que ahora la tabla clientes es la de proveedores se crearán con lo que se denomina metamorfosis en vuelto los objetos de la tabla proveedores automáticamente y en segundos. ¡Simplemente espectacular! ¡Disfrútalo!

48. Objetos visuales. Familias

Utilizando la funcionalidad pegar como creamos los objetos visuales de la tabla de familias a partir de los de clientes. Como las familias es una tabla arbolada con campos diferentes, se resuelven los errores que el inspector muestra en formularios y rejilla. Además, se añade el objeto visor de tabla arbolada y el alternador de lista para poder intercambiar la vista de registros entre el árbol y la rejilla.

49. Objetos visuales. Artículos 1ª parte

Partiendo de los objetos visuales del maestro de clientes, utilizando la opción copiar y pegar como creamos los objetos de artículos. Una vez generados los repasamos ajustándolos a los campos de la tabla de artículos. Comenzando con la rejilla principal de artículos.

50. Objetos visuales. Artículos 2ª parte

Se completan las rejillas de artículos con la de artículos de una familia y selección. A continuación se revisa el formulario de artículos añadiendo los campos de la tabla.

51. Objetos visuales. Artículos 3ª parte

Se completan todos los controles del formulario y el subformulario de descripción e imagen, se ajustan los iconos y se añade la acción a la toolbar principal del menú para a continuación probar la nueva opción de artículos.

52. Objetos visuales. Artículos 4ª parte

Se completa la prueba de grabación de un registro en la tabla de artículos y se corrige el error en el orden de tabulación detectado para el campo código de barras. Ya tenemos listos los objetos para hacer el mantenimiento de la tabla de artículos.

53. Objetos visuales. Botón de menú

Cuando en un formulario existe un control de edición puntero a tabla maestra es habitual añadir un botón de menú a dicho control. Este menú suele tener 3 opciones que permitan al usuario localizar, crear o editar un registro de la tabla maestra. Creamos el menú, las 3 acciones y asociamos el menú al control de edición de forma rápida y sencilla dejándolo preparado para su reutilización con la opción copiar/pegar como.

54. Objetos visuales. Subformulario de un plural

En los formulario de tablas maestras es habitual encontrar en el separador de formularios una pestaña por cada tabla plural de ese maestro. En nuestro caso comenzaremos generando el subformulario de artículos de una familia, para hacerlo copiamos el formulario de familias y lo dejamos sólo con un control vista de datos, que usará un proceso sincronizador de la familia con sus artículos, este proceso devuelve una lista de artículos que se muestran con la rejilla preparada especialmente para mostrar los artículos de una familia. Esta funcionalidad permite al usuario final navegar a través de la información de forma rápida, cómoda y sencilla.

55. Objetos visuales. Facturas de venta 1ª parte

Creamos los objetos visuales de las cabeceras de facturas de venta, partiendo de los objetos de la tabla clientes. Comenzamos ajustando la rejilla con los datos específicos de la factura de venta.

56. Objetos visuales. Facturas de venta 2ª parte

Creamos la rejilla de facturas de venta de un cliente, ajustamos la búsqueda y preparamos el formulario de cabecera de factura de venta.

57. Objetos visuales. Facturas de venta 3ª parte

Preparamos nuestra opción de facturas de venta en el menú, lo probamos y finalmente añadimos el menú de botón de cliente para completar la cabecera de facturas de venta.

58. Objetos visuales. Líneas de detalle de facturas de venta 1ª parte

Una vez que ya tenemos listo el formulario de cabecera de factura de venta llega el turno para desarrollar los objetos visuales que nos servirán para grabar las líneas de detalle de una factura de venta. Utilizamos el formulario y rejilla de artículos para crear los objetos de los movimientos de almacén correspondientes a las líneas de detalle de ventas y ajustamos las columnas de la rejilla de líneas de detalle de venta.

59. Objetos visuales. Líneas de detalle de facturas de venta 2ª parte

Completamos los campos de la rejilla general de líneas de venta y preparamos la rejilla específica para la grabación de líneas de detalle de facturas de venta que tendrá como características especial el ser una rejilla editable que permitirá añadir o modificar líneas de detalle de factura directamente sobre la propia rejilla.

60. Objetos visuales. Líneas de detalle de facturas de venta 3ª parte

Preparamos el subformulario donde se mostrarán las líneas de detalle de la factura y ejecutamos la aplicación comprobando que funciona correctamente la grabación de los movimientos de venta de una factura.

61. Objetos visuales. Líneas de detalle de facturas de venta 4ª parte

Completamos nuestra rejilla de edición incluyen un menú de botón para la columna del artículo donde también se incluye la rejilla de vista de datos para facilitar la selección.

Hemos añadido también varios registros en la tabla de artículos para tener datos de ejemplo a usar en las pruebas de los objetos visuales a desarrollar.

Fichero comprimido con las fotos e iconos usados en el curso:

<http://dl.velneo.com/es/servicios/for...>

62. Objetos visuales. Líneas de detalle de facturas de venta 5ª parte

Creamos el formulario de línea de venta que nos permitirá eliminar registros desde el botón eliminar de la toolbar y también con la tecla suprimir desde la rejilla. Este formulario también permite editar los registros de detalle de una factura y se muestra en cuadro de diálogo ya que no contendrá subformularios y se trata de un formulario de dimensiones reducidas. Se completan así los objetos visuales de la tabla de movimientos de almacén para las líneas de detalle de ventas.

63. Objetos visuales. Facturas de compra

Aprovechando los objetos visuales de facturas y líneas de detalle de venta utilizando copiar y pegar como generamos en segundos los objetos visuales de las facturas y líneas de detalle de compra. Repasamos los objetos ajustando las diferencias entre las ventas y las compras como es el cambio de cliente por proveedor y añadimos la opción de facturas de compra a la toolbar general.

64. Objetos visuales. Líneas de detalle de facturas de compra

Para completar la grabación de las facturas de compra se revisan los

objetos visuales de líneas de detalle de facturas de compra y se ejecuta la aplicación para comprobar que ya están totalmente operativas las facturas de compra.

65. Objetos visuales. Subformularios de plurales de facturas de compras y ventas

Una vez que ya tenemos creados los objetos de facturas de compras y ventas ha llegado el momento de crear los subformularios de plurales de facturas de venta de un cliente y facturas de compra de un proveedor, lo que nos permite consultar esa información además de crear o editar facturas de venta o compra directamente desde la pestaña de un cliente o proveedor, respectivamente, aprovechando incluso la inducción automática del maestro.

66. Objetos visuales. Subformularios de plurales de ventas de un cliente

Seguimos completando la interfaz de nuestra aplicación y ahora toca el turno de incluir en el formulario de clientes la pestaña que nos permita consultar el detalle de las ventas de un cliente, aumentado de esta forma las posibilidades de navegación por la información del usuario final.

67. Objetos visuales. Subformulario de plural de movimientos de un artículo 1ª parte

Para incluir en el formulario de un artículo una pestaña con el detalle de todos los movimientos de almacén hemos mejorado la tabla de movimientos incluyendo dos campos fórmula que devuelven el nombre del cliente o proveedor y el segundo campo el nº factura de venta o compra, en función de si el tipo de movimientos es de venta o compra. Con esa información hemos dejado lista un rejilla que nos servirá para mostrar el detalle de los movimientos de almacén de un artículo.

68. Objetos visuales. Subformulario de plural de movimientos de un artículo 2ª parte

Preparamos el subformulario de movimientos de almacén de un artículo partiendo del de ventas de un cliente. Probamos lo desarrollado y observamos que podemos mejorar la rejilla de movimientos de almacén para que muestre la cantidad y el importe en negativo en función de si el movimiento es de compra o venta, así que procedemos a mejorar la tabla de movimientos de almacén con nuevos campos fórmula, mejoras en los índices de movimientos por artículo y cliente.

69. Objetos visuales. Subformulario de plural de movimientos de un artículo 3ª parte

Procedemos a ajustar la rejilla de movimientos de almacén de un artículos ajustado el alta del cuerpo ahora que hemos quitado la foto y el contenido de las columnas cantidad e importe con los nuevos campos fórmula. También mejoramos la ordenación aplicando el comando invertir lista en la carga de los plurales de movimientos de un artículo y de un cliente y verificamos en ejecución que todo funciona correctamente.

70. Objetos visuales. Descuentos por cliente y familia 1ª parte

Creamos los objetos visuales de la tabla de descuentos por cliente y familia, partiendo de los objetos visuales de la tabla de facturas de venta ya que tiene el campo puntero a cliente resuelto y lo podemos aprovechar. Una vez generados los objetos repasamos la rejilla principal y creamos las rejillas específicas de descuentos por familia de un cliente y de descuentos por cliente de una familia, ambas editables y por último también preparamos los botones de menú para los campos cliente y familia.

71. Objetos visuales. Descuentos por cliente y familia 2ª parte

Aplicamos los botones de menú en las rejillas editables en las columnas

correspondientes al cliente y la familia. Después preparamos el formulario de descuento por cliente y familia que puede ser utilizado para altas, modificaciones y bajas de descuentos por cliente y familia. Comenzamos a preparar el subformulario de descuentos por familia de un cliente.

72. Objetos visuales. Descuentos por cliente y familia 3ª parte

Para terminar se crean los subformularios de descuentos por familia de un cliente y de descuentos por cliente de una familia. Finalmente se ejecuta la aplicación, se ajustan algunos alineamientos del formulario de descuentos así como que se muestre en cuadro de diálogo y se verifica el correcto funcionamiento de la navegación a través de estos nuevos objetos visuales.

73. Objetos visuales. Estadísticas por cliente y artículo 1ª parte

Creamos las rejillas de la tabla de estadísticas que usaremos para visualizarlas en pestañas de los formularios de clientes y familias. En las rejillas se muestran todos los campos numéricos calculados como las ventas, costes, beneficio y % beneficio.

74. Objetos visuales. Estadísticas por cliente y artículo 2ª parte

Creamos los subformularios de estadísticas por artículo y año de un cliente y de estadísticas por cliente y año de un artículo y añadimos estos subformularios en las pestañas de los formularios de cliente y artículo respectivamente. Finalmente ejecutamos la aplicación para verificar que las estadísticas se muestran correctamente en las pestañas de estadísticas de los formularios de cliente y artículo.

75. Objetos visuales. Informe de factura de venta 1ª parte

Creamos el objeto informe con origen de movimientos de almacén ya que desde esa tabla tenemos acceso a toda la información del detalle y de la cabecera y datos del cliente gracias a los enlaces permanentes. Creamos

las secciones de cabecera de página, detalle pie de informe. Comenzamos a añadir controles a la sección de cabecera como el logotipo del cliente y algunos textos.

76. Objetos visuales. Informe de factura de venta 2ª parte

Completamos los datos de la sección de cabecera donde se incluyen el nº factura, la fecha factura, el nombre y CIF del cliente y los textos de cabecera de las columnas del detalle.

77. Objetos visuales. Informe de factura de venta 3ª parte

Completamos la cabecera, el detalle y comenzamos a poner en el pie los controles para totalizar la cantidad y el importe total factura.

78. Objetos visuales. Informe de factura de venta 4ª parte

Completamos el informe añadiendo una variable local para calcular el total cantidad y ajustamos los controles del pie. Creamos un objeto alternador que nos permitirá seleccionar entre la rejilla de líneas de detalle de la factura y el nuevo informe.

79. Objetos visuales. Informe de factura de venta 5ª parte

Finalizamos nuestro informe sustituyendo la rejilla del subformulario de líneas de detalle de factura de venta por el nuevo alternador que debe incluir la toolbar que se debe quitar de las rejillas incluidas en el alternador para que no se dupliquen. Tras una primera prueba se revisan las toolbar de las rejillas para que no salgan duplicadas y añade el subobjeto de cálculo del total cantidad quedando nuestro informe listo para ser mostrado en presentación preliminar, pudiendo el cliente final desde la toolbar de la presentación preliminar imprimirlo a papel o generar un fichero en disco con el informe en formato PDF.

80. Testeo de la aplicación desarrollada

Organizamos las opciones de la toolbar principal y repasamos todas las historias de usuario que nos solicitó el cliente final y que hemos cubierto con esta aplicación desarrollada en un tiempo récord.

81. Gracias, gracias, gracias por completar el curso

Como dice el título, desde Velneo queremos darte las GRACIAS con mayúsculas por haber completado este curso.

Esperamos que hayas aprendido mucho y hayas conseguido el nivel necesario para comenzar a desarrollar tus primeras aplicaciones con Velneo.

En nuestro [canal de formación](#) encontrarás multitud de recursos para seguir ampliando tu formación con niveles más avanzados.

Deseamos que hayas disfrutado con el curso tanto como nosotros durante su preparación y que hayas encontrado en Velneo tu aliado perfecto para desarrollar grandes aplicaciones.

Gracias por acompañarnos y haber realizado el gran esfuerzo de completar este curso.