



# PRESENTACIÓN DE PROYECTO

Control de Calidad del Software



GERARDO ABRAHAM LIZBETH ÁNGEL

# ¿QUIÉNES SOMOS?



## Sobre Nosotros

- GALA Developer Group es una empresa joven dedicada a ofrecer soluciones de e-commerce, servicios de marketing digital, aplicaciones web y software de escritorio.



## Misión

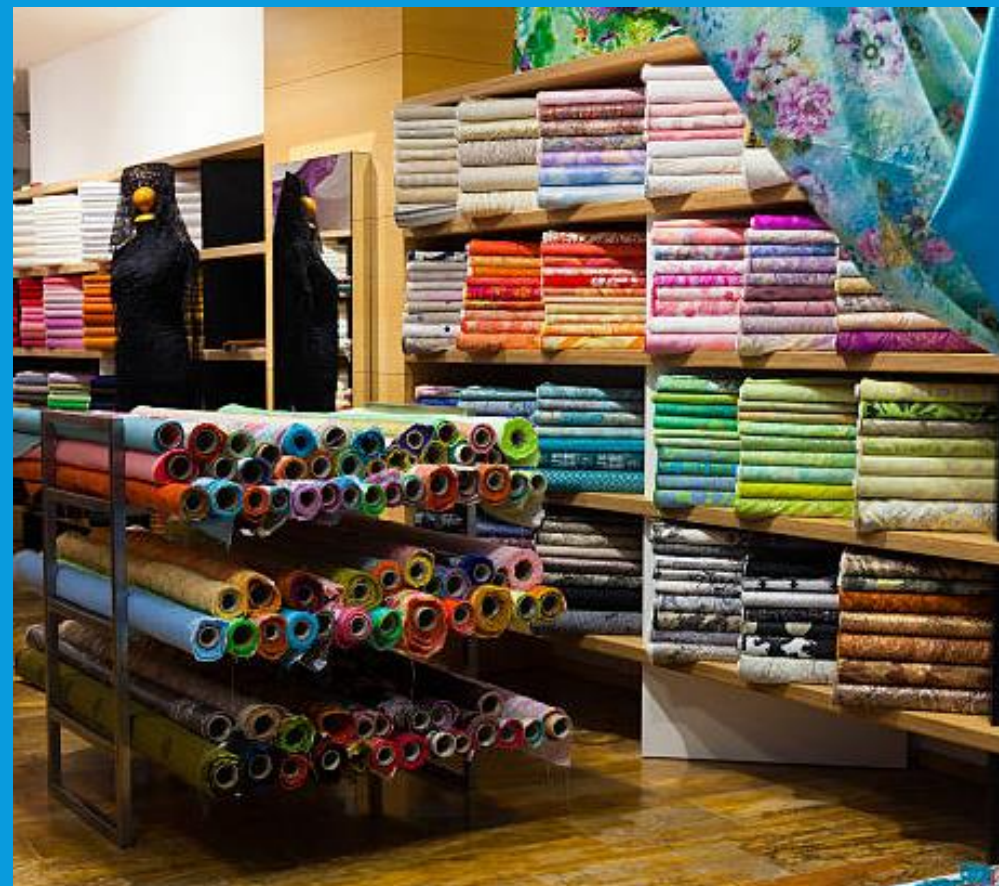
- Brindar soluciones de desarrollo de software, ingeniería web, soluciones informáticas y soluciones tecnológicas integrales con el compromiso en la calidad, construyendo relaciones comerciales a largo plazo, mejorando y optimizando el proceso de negocio y competitividad de nuestros clientes.

## Visión

- Convertirnos en una empresa consolidada de reconocido prestigio a nivel estatal y nacional, brindando soluciones especializadas en el área de informática, tecnología y software.

# INTRODUCCIÓN

- Debido a que se trata de un sistema para un negocio pequeño, ya que habrá un único usuario vendedor. Los cobros se llevaran a cabo de manera personal, lo único que registrará el sistema será los precios de los artículos vendidos.
- El sistema será útil para la persona que crea y venda artículos tejidos porque le va a permitir tener un control de los productos que ha vendido, compartir ejemplos de diseño para los clientes, administrar a los clientes y proporcionar un tiempo estimado de entrega.
- Este sistema sólo funcionará ejecutándolo a través del framework QT.



# COTIZACIÓN GENERAL

Servicio		Costo estimado	Tiempo estimado
Diseño	Base de datos	\$300	1 día
	Interfaces	\$3,600	2 días
	Realizar pedido	\$3,656.25	5 días
Programación	Administrar pedidos	\$3,412.50	5 días
	Administrar clientes	\$3,900	5 días
	Administrar artículos	\$5,362.50	6 días
Implementación de la aplicación(extra)		\$2,000	1 día
Utilidades		\$1,788	-
Total		\$24, 019.25 MXN	25 días

# ESTIMACIÓN DE COSTES

- Para determinar la estimación de costos del software a desarrollar primero elegimos un método para medir su tamaño. Para este proyecto se seleccionó el método de análisis de puntos de función con COSMIC.
- El proyecto es compuesto por los siguientes procesos funcionales y movimiento de datos:

Procesos funcionales y movimiento de datos	Puntos de función
Realizar pedido	15CFP
Administrar pedidos:	14CFP
Administrar clientes:	16CFP
Administrar artículos:	22CFP
<b>Tamaño del proyecto</b>	<b>67CFP</b>

- De acuerdo con el equipo de desarrollo necesario para la producción de este proyecto sabemos que el costo mensual será de \$19,500.

<ul style="list-style-type: none"><li>• Costo por punto de función = Costo mes del equipo de trabajo / puntos de función del mes</li></ul> <p>Costo por punto de función = <b>\$ 19,500 MXN / 80 puntos de función =</b> <b>\$ 243.75 MXN / Punto de función</b></p>	<ul style="list-style-type: none"><li>• Costo del proyecto de software = <b>67 CFP x \$ 243.75 MXN</b></li></ul> <p>Costo del proyecto de software = <b>\$ 16,331.25 MXN</b></p>	<ul style="list-style-type: none"><li>• Duración del proyecto = <b>67 puntos de función COSMIC / 80 puntos de función COSMIC mes.</b></li></ul> <p><b>Duración del proyecto = 0,8375 meses</b></p>
--	--	--

# JUSTIFICACIÓN DE LOS COSTOS

- Costos directos

Cantidad de personal	Personal	Número de meses	Salario mensual	Total
1	Programador/ Analista C++	.8375	\$15, 000	\$12, 000
1	Diseñador gráfico	1/2	\$ 7,200	\$3,600
			\$19, 600 (por mes)	<u>Total: \$15, 600</u>

# JUSTIFICACIÓN DE LOS COSTOS

- Costos indirectos

Cantidad	Productos	Costos mensuales	Duración	Costal Total
1	Luz	\$750	1 meses	\$750
1	Agua	\$250	1 meses	\$250
1	Teléfono e Internet	\$450	1 meses	\$450
24	Garrafón de agua	\$240	1 meses	\$240
-	Presupuesto para papelería (paquete de 500 hojas)	\$98	1 meses	\$98
-(No aplica)	Presupuesto para limpieza (jabón, gel, etc.)	\$250	1 meses	\$250
-(No aplica)	Renta de oficina central	\$2500	1 meses	\$2500
Total:				<b>\$1788</b>



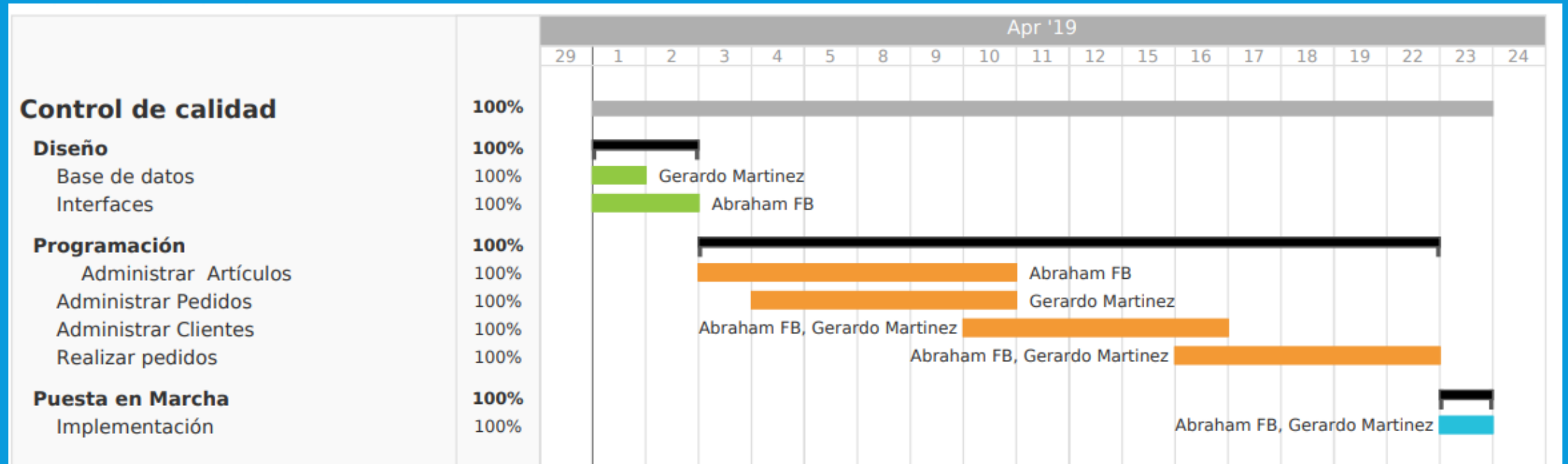
# JUSTIFICACIÓN DE LOS COSTOS



## • Costo de venta al cliente

El costo para la empresa desarrolladora fue de \$24, 019.25 MXN, tomando en cuenta las bases de negocio de la empresa las ganancias constan del 10% del costo total del proyecto siendo las ganancias de \$2, 401.92 MXN, por tanto, el costo final de venta al cliente sería de \$26, 412.17 MXN.

# DIAGRAMA DE GANTT



# METODOLOGÍA RUP

- Es un proceso de la ingeniería de Software que proporciona un acercamiento disciplinado a la asignación de tareas y responsabilidades dentro de un proyecto. Su propósito es asegurar la calidad del software al punto de que cumpla las necesidades de los usuarios finales, con unos costos y calendario predecibles.

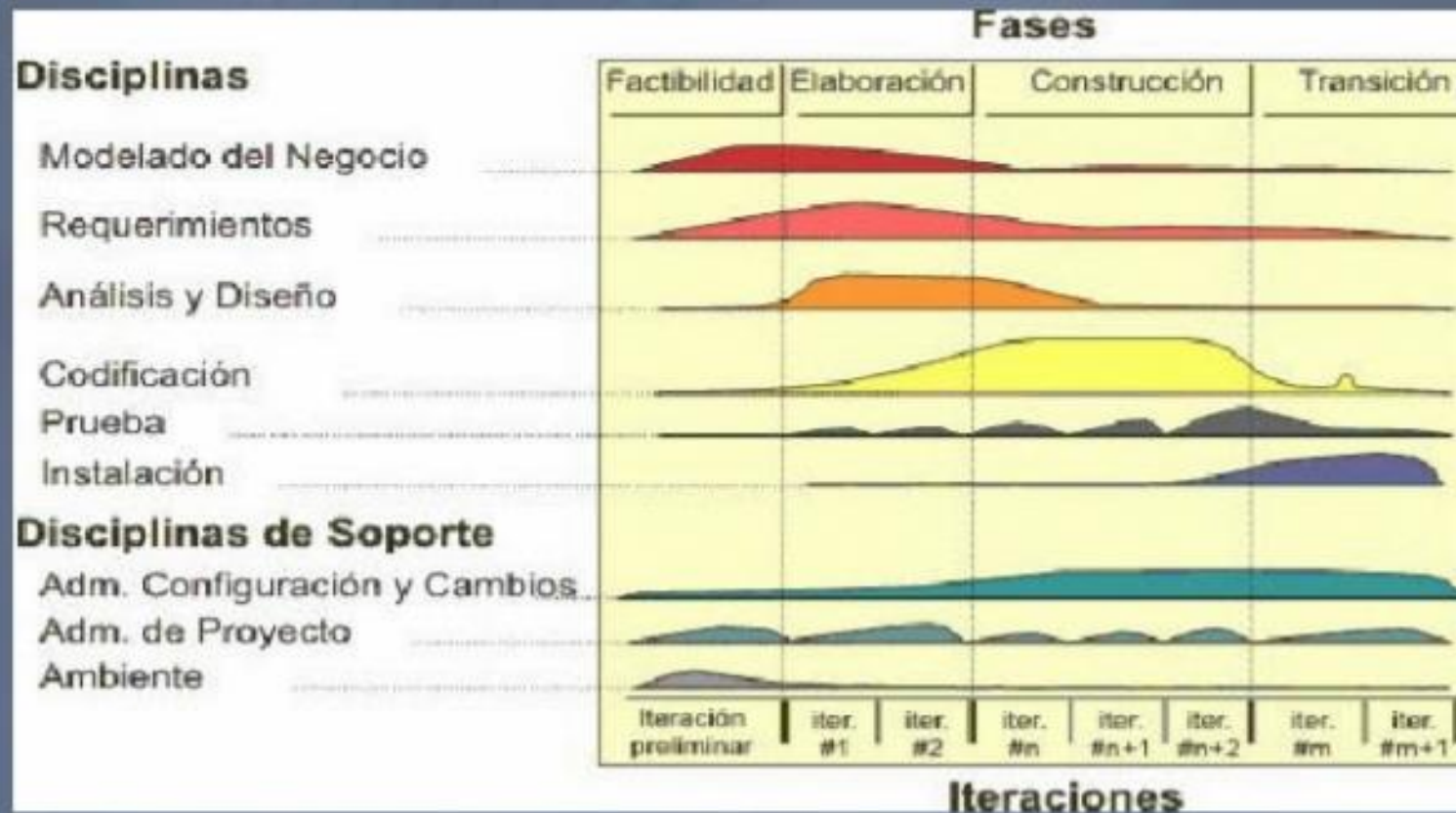


# FASES DE RUP

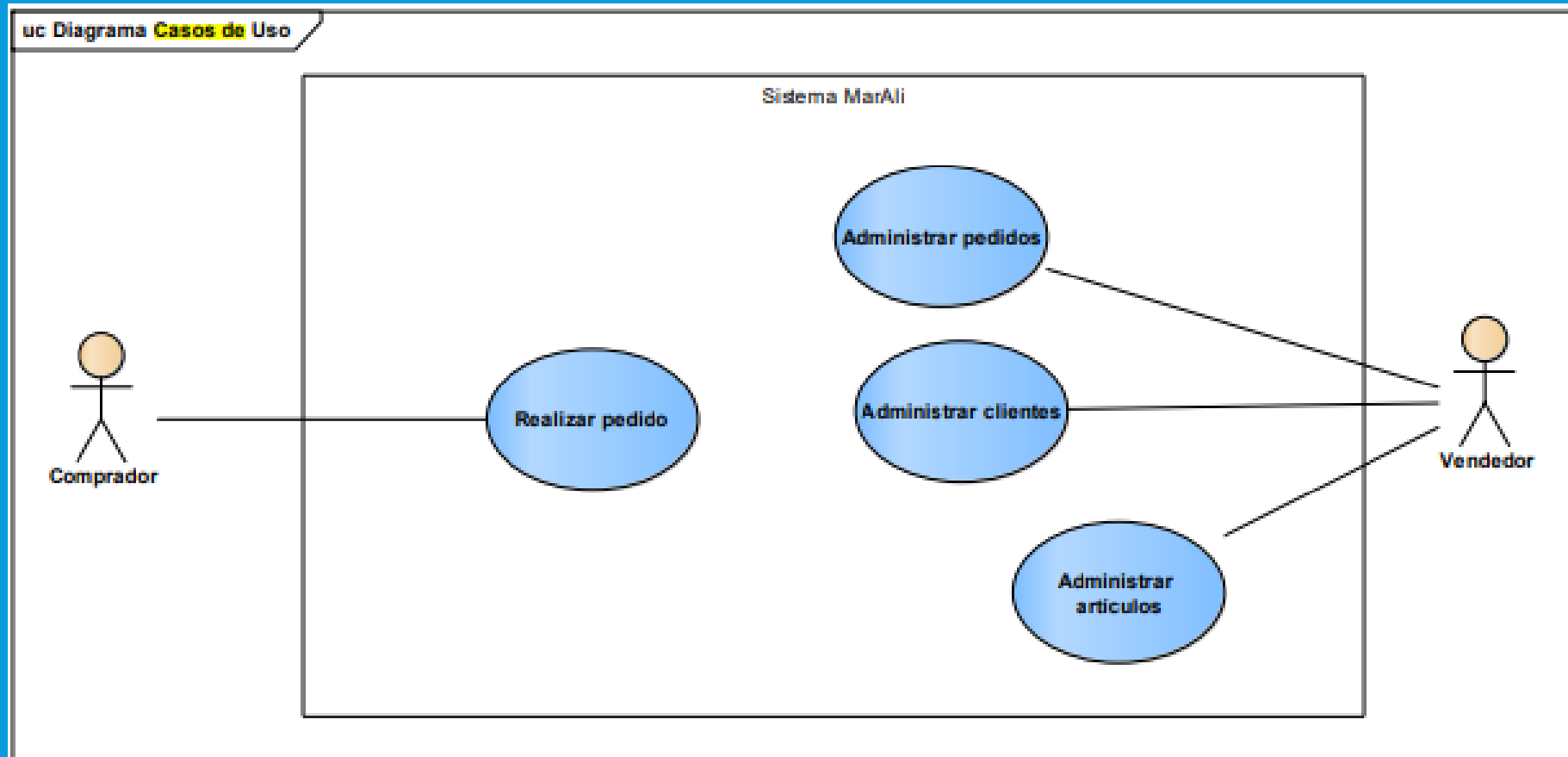
- **Fase de inicio:** Define el alcance del proyecto
- **Fase de elaboración:** Planificación del proyecto, elaboración de una arquitectura base.
- **Fase de construcción:** Construir el sistema.
- **Fase de transición:** Transición a usuarios.

# FASES E ITERACIONES

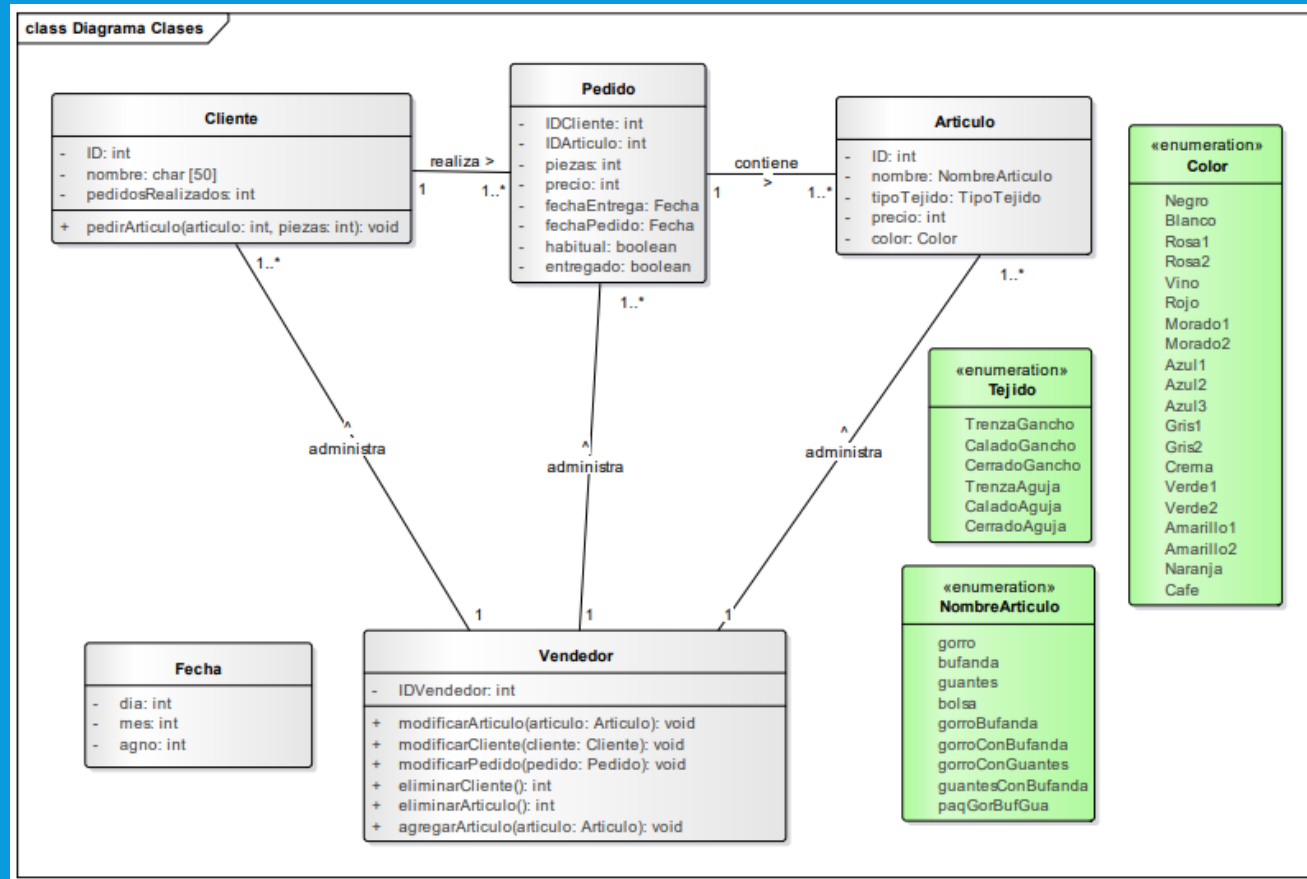
## Disciplinas Básicas



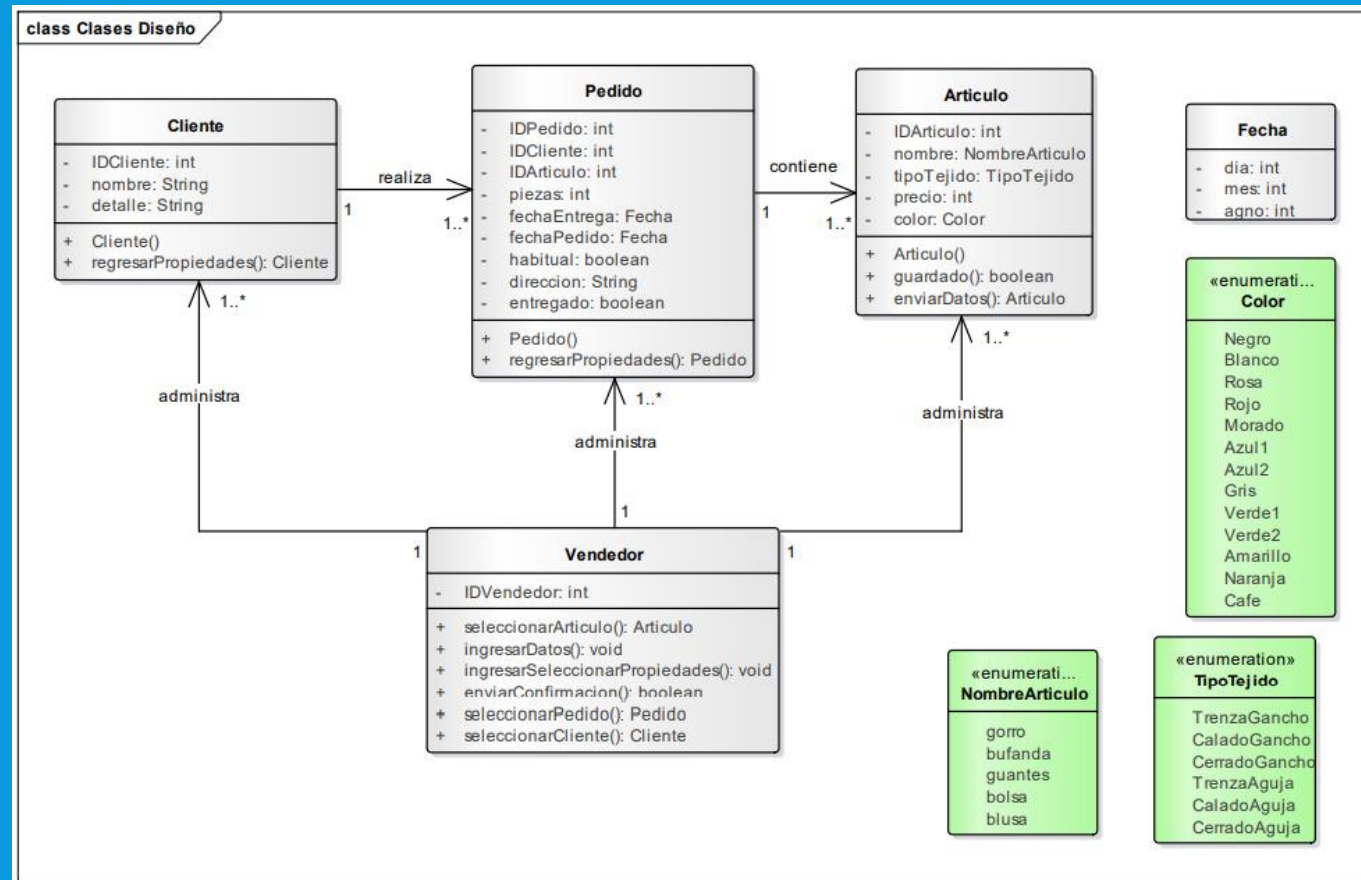
# DIAGRAMA DE CASOS DE USO



# DIAGRAMA DE CLASES



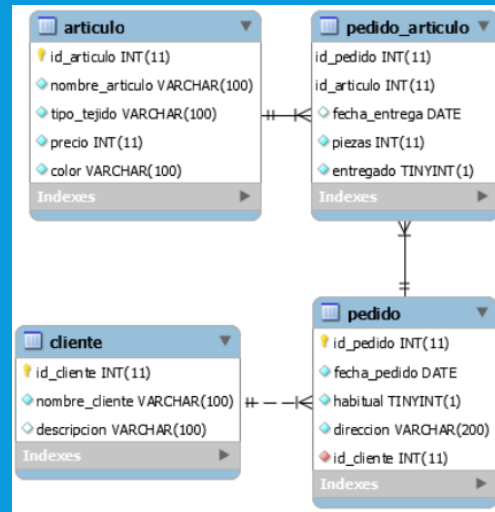
# DIAGRAMA DE CLASES DE DISEÑO PRELIMINAR





# DISEÑO DE LA BASE DE DATOS

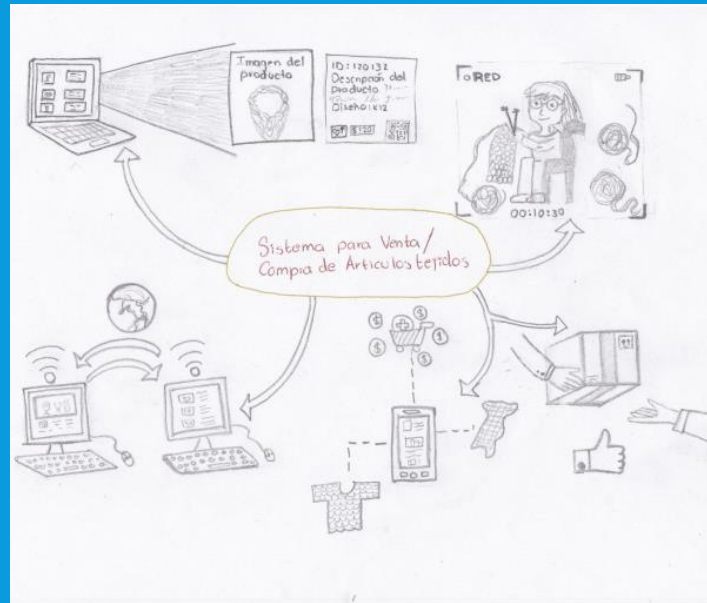
De acuerdo con el diagrama de clases, se necesita almacenar los siguientes campos en una Base de Datos:



Por lo tanto, resulta el siguiente diagrama de la Base de Datos (no normalizada) intentando ubicar adecuadamente cada campo en la tabla correspondiente.

# TÉCNICAS DE RECOPILOACIÓN DE INFORMACIÓN.

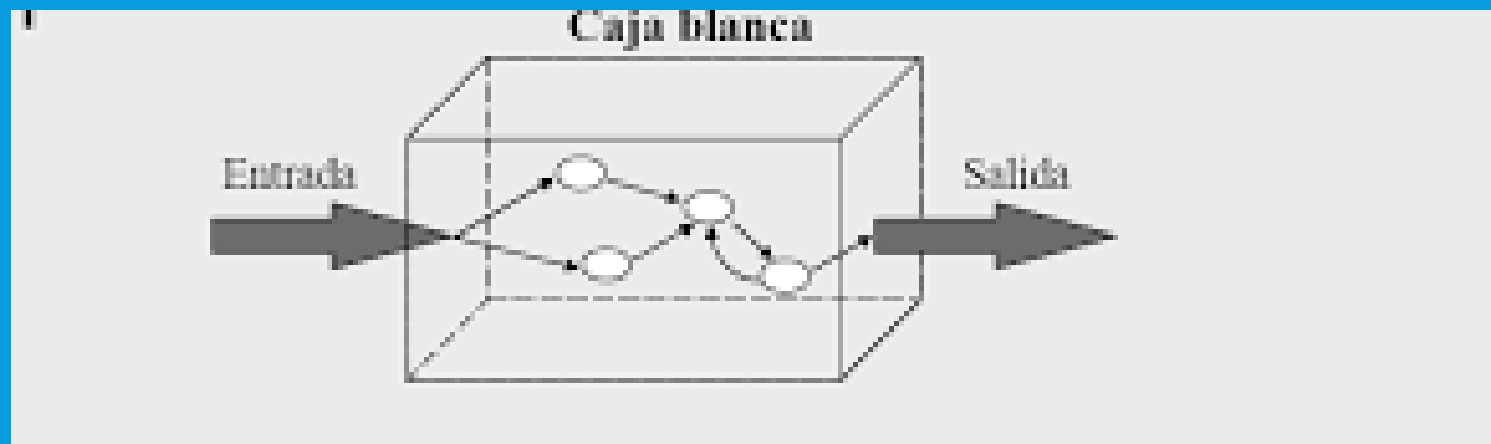
- Es útil para conocer cómo visualiza el sistema una persona ajena al contexto de computación en base a la descripción realizada. Es parte del MODIHC.



Mapa mental del sistema realizado por una estudiante de Urbanismo y Diseño Ambiental.

# PRUEBAS DE CAJA BLANCA I

- Se centran en los detalles procedimentales del software, por lo que su diseño está fuertemente ligado al código fuente, lo cual, esto significa que tenemos que realizar un “estudio”, o más bien, un análisis de nuestro código, para esto, existen tres pautas fundamentales para poder realizar con éxito una prueba de caja blanca.



# PRUEBAS DE CAJA BLANCA II

Nombre de los Autores del Caso de Prueba:		Realizar pedido	
Fecha de Elaboración: Tarea o Función Selección de pedido habitual	08 de abril de 2019		
	Entrada	Prueba	Salida
	Id del artículo		Propiedades del artículo
	Cantidad de piezas y dirección		Tabla de artículos finales del pedido
	Id diferente		El sistema muestra un diálogo indicándole al cliente que el producto con el ID que ingresó no existe y le pide que intente con un ID correcto
		Solicitud de artículo	
	Confirmación del cliente		Una notificación de pedido exitoso
	Campos vacíos		El sistema muestra un diálogo pidiéndole al cliente que ingrese todos los campos solicitados
Selección de pedido especial	Selección o ingreso de los datos del artículo		Tabla de los artículos especiales
	Confirmación del pedido	Elección de pedido	Notificación de pedido exitoso
	Campos vacíos		Se muestra una notificación de si desea seguir si ordenar algún pedido especial

# PRUEBAS DE CAJA BLANCA III

Nombre de los Autores del Caso de Prueba:		Administrar pedidos	
Fecha de Elaboración:	08 de abril de 2019		
Tarea o Función	Entrada	Prueba	Salida
Inicio de sesión	Campos vacíos	Entrar al sistema	El sistema permite el acceso a la interfaz del vendedor/Administrador
Encadenamientos alternativos	El vendedor ingresa ID incorrecto		El sistema muestra una advertencia con la leyenda "ID incorrecto"
	Id correcto		Se carga correctamente el pedido
	El vendedor selecciona un cliente que eliminó previamente (no se actualiza la lista)	Errores del vendedor	El sistema no hace ninguna operación
	El vendedor se equivoca al seleccionar un cliente que sí existe en el sistema		El sistema guarda información incorrecta del cliente

# PRUEBAS DE CAJA BLANCA IV

Nombre de los Autores del Caso de Prueba:		Administrar clientes	
Fecha de Elaboración:	o8 de abril de 2019		
Tarea o Función	Entrada	Prueba	Salida
Verificación del id	Id del vendedor	Comportamientos del vendedor	El sistema muestra la lista de los clientes registrados
	El vendedor selecciona un cliente y selecciona la opción “Modificar”.		El sistema arroja un diálogo al vendedor para que modifique los datos que desee
	El vendedor selecciona un cliente		El sistema arroja un diálogo de confirmación.
Encadenamientos alternativos	El vendedor ingresa usuario y/o contraseña incorrectos	Opciones	El sistema muestra una advertencia con la leyenda “ID incorrecto”
Encadenamientos de error (excepciones)	El vendedor selecciona un artículo que eliminó previamente (no se actualiza la lista)	Errores del vendedor	El sistema no hace ninguna operación
	El vendedor se equivoca al seleccionar un artículo que sí existe en el sistema		El sistema guarda información incorrecta del artículo, por lo tanto, el vendedor tiene que modificar de nuevo, pero si no recuerda los datos anteriores, se pierden

# PRUEBAS DE CAJA BLANCA V

Nombre de los Autores del Caso de Prueba:		Administrar artículos	
Fecha de Elaboración:	08 de abril de 2019		
Tarea o Función	Entrada	Prueba	Salida
Inicio e ir a panel	id de usuario	El vendedor entra en el apartado “Artículos->Modificar/Eliminar”	El sistema muestra la lista de los artículos publicados como ejemplo para los clientes
Modificar	El sistema arroja un diálogo al vendedor para que modifique los datos que desee	Modificación del artículo	El sistema guarda los cambios con éxito.
Agregar artículo	El administrador ingresa los datos para registrar un nuevo artículo.	Agregar artículo	El artículo se agrega con éxito en el sistema.
Eliminar artículo	El administrador selecciona un artículo (que no está asociado con un pedido) para eliminarlo.	Eliminar artículo	El artículo se elimina con éxito.
	El administrador selecciona un artículo (que está asociado con un pedido) para eliminarlo.		Se muestra un dialogo que informa que el artículo no puede ser eliminado.

# PRUEBAS DE CAJA BLANCA (INICIAR SESIÓN)

- Entrada: Campos vacíos

```
void MainWindow::on_pushButtonCliente_clicked()
{
    QString passwInicio;
    int passwInicioInt;
    passwInicio=ui->lineEdit_idIngresar->text();
    if(passwInicio.isEmpty()){
        ui->stackedWidget->setCurrentIndex(1);
        ui->tabWidget->setCurrentIndex(0);
        Modelo3->setFilter("nombre_cliente LIKE '%+%')");
        ui->ArticulosPedidosTabla->setModel(Modelo3);
    }
    else{
        passwInicioInt=passwInicio.toInt();
        if(passwInicioInt!=3102){
            QMessageBox::information(this,"ID incorrecto","ID de administrador incorrecto."
                                     " Para ingresar como cliente,"
                                     " sólo presione 'Ingresar'.");
        }
        else{
            ui->stackedWidget->setCurrentIndex(2);
            ui->tabWidget_2->setCurrentIndex(0);
            ui->radioButton1->clicked(true);
        }
    }
    ui->lineEdit_idIngresar->clear();
}
```



# PRUEBAS DE CAJA BLANCA (INICIAR SESIÓN)

- Entrada: Id incorrecto

```
void MainWindow::on_pushButtonCliente_clicked()
{
    QString passwInicio;
    int passwInicioInt;
    passwInicio=ui->lineEdit_idIngresar->text();
    if(passwInicio.isEmpty()){
        ui->stackedWidget->setCurrentIndex(1);
        ui->tabWidget->setCurrentIndex(0);
        Modelo3->setFilter("nombre_cliente LIKE '%+%')");
        ui->ArticulosPedidosTabla->setModel(Modelo3);
    }
    else{
        passwInicioInt=passwInicio.toInt();
        if(passwInicioInt!=3102){
            QMessageBox::information(this,"ID incorrecto","ID de administrador incorrecto."
                                     " Para ingresar como cliente,"
                                     " sólo presione 'Ingresar'.");
        }
        else{
            ui->stackedWidget->setCurrentIndex(2);
            ui->tabWidget_2->setCurrentIndex(0);
            ui->radioButton1->clicked(true);
        }
    }
    ui->lineEdit_idIngresar->clear();
}
```

# PRUEBAS DE CAJA BLANCA (INICIAR SESIÓN)

- Entrada: Id correcto

```
void MainWindow::on_pushButtonCliente_clicked()
{
    QString passwInicio;
    int passwInicioInt;
    passwInicio=ui->lineEdit_idIngresar->text();
    if(passwInicio.isEmpty()){
        ui->stackedWidget->setCurrentIndex(1);
        ui->tabWidget->setCurrentIndex(0);
        Modelo3->setFilter("nombre_cliente LIKE '%+%')");
        ui->ArticulosPedidosTabla->setModel(Modelo3);
    }
    else{
        passwInicioInt=passwInicio.toInt();
        if(passwInicioInt!=3102){
            QMessageBox::information(this,"ID incorrecto","ID de administrador incorrecto."
                                     " Para ingresar como cliente,"
                                     " sólo presione 'Ingresar'.");
        }
        else{
            ui->stackedWidget->setCurrentIndex(2);
            ui->tabWidget_2->setCurrentIndex(0);
            ui->radioButton1->clicked(true);
        }
    }
    ui->lineEdit_idIngresar->clear();
}
```

# PRUEBAS DE CAJA NEGRA

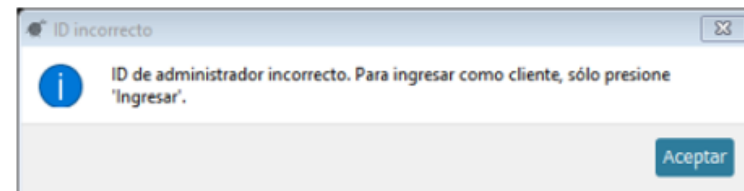
Es una técnica de pruebas de software en la cual la funcionalidad se verifica sin tener en cuenta la estructura interna de código (a diferencia de la prueba de caja blanca), detalles de implementación o escenarios de ejecución internos en el software.



# PRUEBAS DE CAJA NEGRA I

## Caso 1:

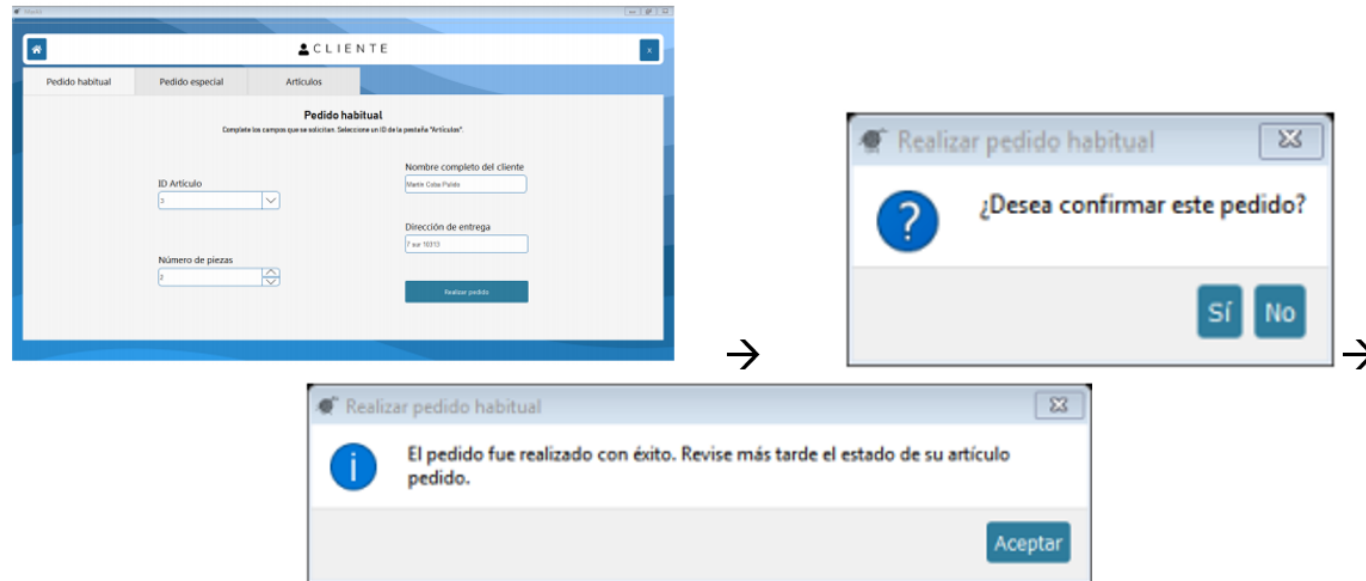
- Datos de entrada: id no registrado.
- Resultado esperado (salida): la aplicación no arroja un error de id incorrecto o en su defecto ingresar como cliente.



# PRUEBAS DE CAJA NEGRA II

## Caso 2:

- Datos de entrada: un conjunto de datos del pedido.
- Resultado esperado (salida): la aplicación nos muestra una ventana de confirmación del pedido.



# PRUEBAS DE CAJA NEGRA III

## Caso 3:

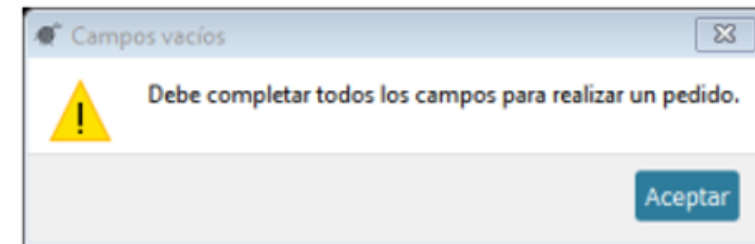
- Datos de entrada: nada (campo de texto vacío).
- Resultado esperado (salida): la aplicación nos arroja un error donde nos pide que completemos los campos del pedido.



The screenshot shows a web application interface for a client. At the top, there's a header with a logo and the word 'CLIENTE'. Below the header, there are three tabs: 'Pedido habitual', 'Pedido especial', and 'Artículos'. The 'Pedido habitual' tab is selected. The form contains the following fields:

- ID Artículo:** A dropdown menu with the value '1' selected.
- Número de piezas:** A spinner box with the value '1'.
- Nombre completo del cliente:** An empty text input field.
- Dirección de entrega:** An empty text input field.
- Realizar pedido:** A blue button at the bottom right.

Below the tabs, there's a sub-header 'Pedido habitual' and a note: 'Complete los campos que se indican. Seleccione un ID de la pestaña "Artículos".'



# PRUEBAS DE CAJA NEGRA IV

## Caso 4:

- Datos de entrada: nada en los campos de nombre o dirección de entrega (campos vacíos).
- Resultado esperado (salida): la aplicación nos muestra que debemos completar los campos de “nombre del cliente” y “dirección de entrega”.

CLIENTE

Pedido habitual Pedido especial Artículos

**Pedido especial**  
Complete los campos que se solicitan. Puede pedir 1 o 2 artículos.

Artículo 1

Nombre:

Tipo de tejido:

Color:

Número de piezas:

Artículo 2 (Opcional)

Nombre:

Tipo de tejido:

Color:

Número de piezas:

Nombre completo del cliente:

Dirección de entrega:

Realizar pedido



Campos vacíos

⚠ Debe completar los campos de 'nombre del cliente' y 'dirección de entrega' para realizar un pedido.

Aceptar

# PRUEBAS DE CAJA NEGRA V

## Caso 5:

- Datos de entrada: campos vacíos en el segundo artículo o pedido especial.
- Resultado esperado (salida): la aplicación nos muestra un mensaje de advertencia de que no va a pedir un segundo o artículo especial, en caso de que lo requiera deben llenarse obligatoriamente.

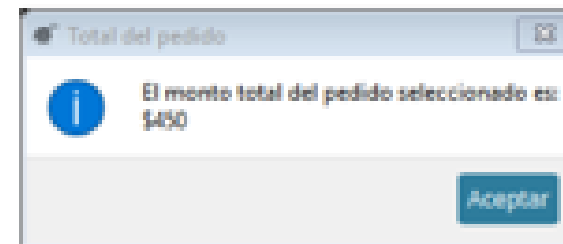
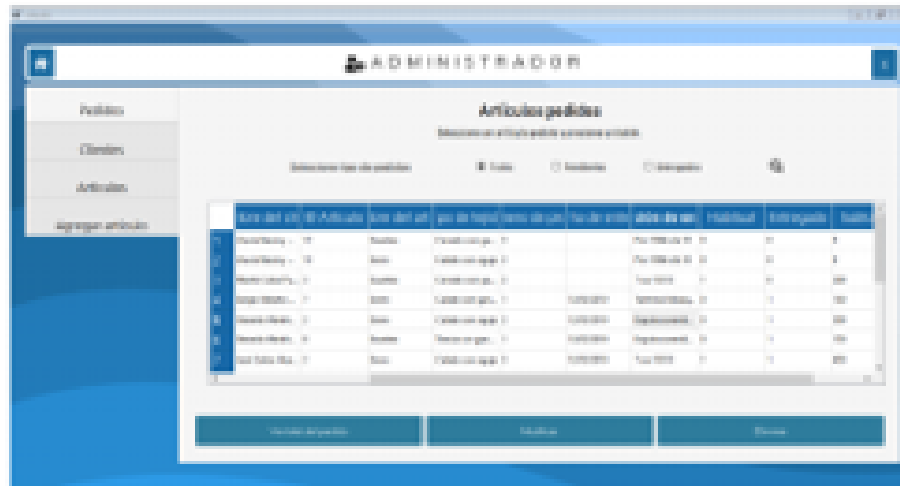
The screenshot shows a web application interface for a customer ('CLIENTE'). It has three tabs: 'Pedido habitual', 'Pedido especial', and 'Artículos'. The 'Pedido especial' tab is active, showing a form to complete fields for two articles. Article 1 has fields for Name, Type of fabric, Color, and Number of pieces. Article 2 (Optional) has the same fields. There are also fields for the customer's full name and delivery address. A 'Realizar pedido' button is at the bottom right of the form. A warning message box titled 'Artículo incompleto' is overlaid on the right, with a yellow warning icon and the text: 'El segundo es opcional. Deje vacíos todos los campos de este artículo si no desea pedirlo o complete todos si desea pedirlo.' An 'Aceptar' button is at the bottom right of the message box.



# PRUEBAS DE CAJA NEGRA VI

### Caso 6:

- Datos de entrada: selección del botón “ver total del pedido”.
- Resultado esperado (salida): la aplicación nos devuelve el total del pedido.



# PRUEBAS DE CAJA NEGRA VII

## Caso 7:

- Datos de entrada: selección de fecha en el calendario.
- Resultado esperado (salida): la aplicación nos lanza una advertencia de que "no se puede cambiar alguna fecha de entrega si ya existía una anterior a la actual".

MODIFICAR ARTÍCULO PEDIDO

Seleccione una fecha en el calendario para cambiar la fecha de entrega del artículo pedido.

mayi, 2019

dom.	lun.	mar.	mié.	jue.	vie.	sáb.
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

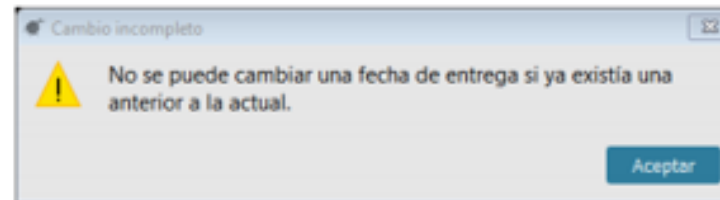
ID Pedido:

ID Artículo:

Fecha de entrega:

Entregado:

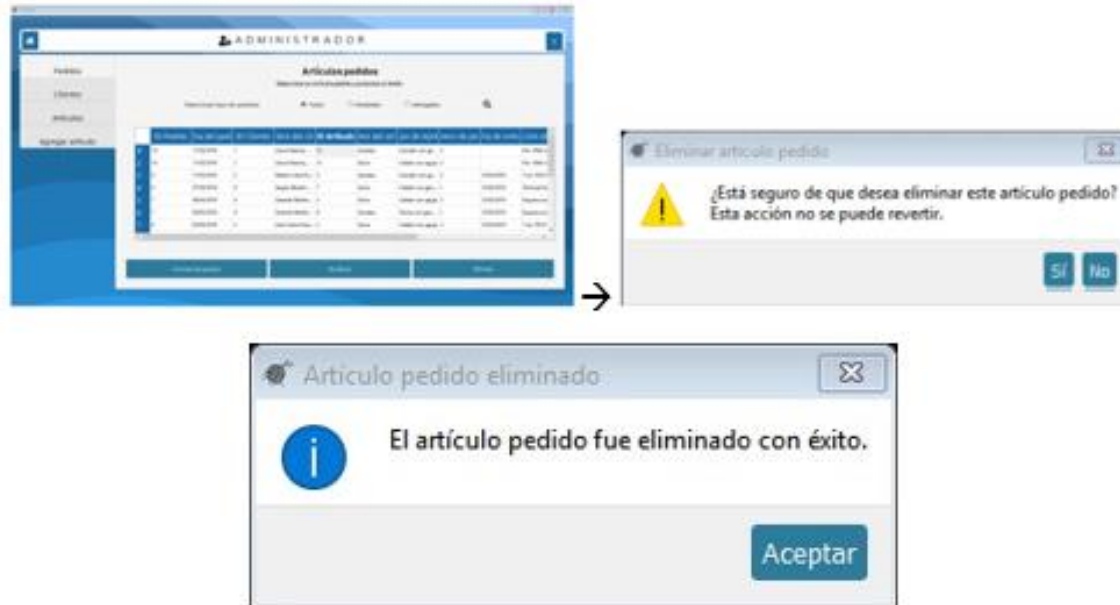
Aceptar Cancelar



# PRUEBAS DE CAJA NEGRA VIII

## Caso 8:

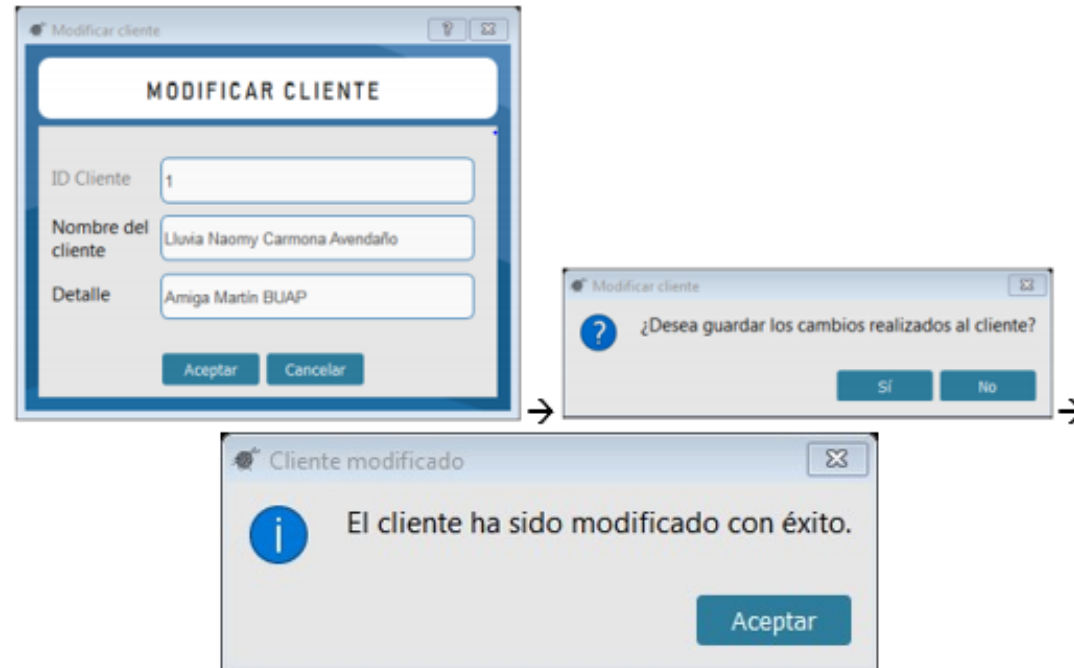
- Datos de entrada: selección del botón “eliminar”.
- Resultado esperado (salida): la aplicación nos pedirá que confirmemos y posteriormente nos mostrará un mensaje de advertencia que esta acción no se puede revertir.



# PRUEBAS DE CAJA NEGRA IX

Caso 9:

- Datos de entrada: datos nuevos del cliente.
- Resultado esperado (salida): la aplicación nos mostrará una ventana de confirmación de la información.



# GALA DEVELOPER GROUP

*¡Gracias!*

Contacto: 712 1069 839

Sitio web: [gala-dev.com](http://gala-dev.com)

