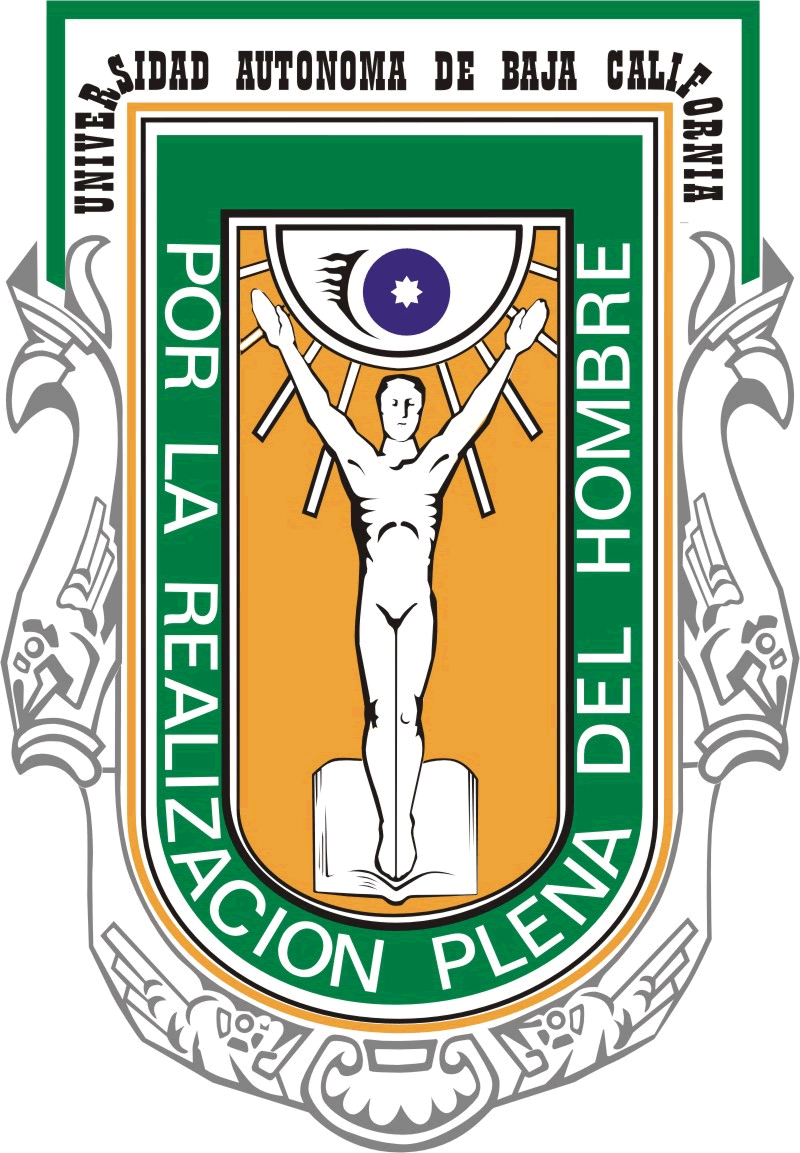
Universidad Autónoma de Baja California

Facultad de Ciencias Químicas e Ingeniería



**Profesora:** Olivia Mendoza Duarte

**Materia:** Base de Datos

**Grupo:** 561

**Proyecto:** Sistema para control de inventarios

**Integrantes:**

Morales Rosales Iván Alfredo 1231098

Jaquez Galvan Jesus Alberto 1238268

Tijuana, Baja California a 30 de Mayo del 2019

Índice

1.Introducción………………………………………………………………..2

2.Descripción de la Problemática…………………………..…..2

3.Requerimientos…………………………………………………………..3

4.Desarrollo e implementación…………………………………...4

5.Procedimientos Almacenados…………...……….……….….5

5.1.Procedimiento 1………………………………….…….….5

5.2.Procedimiento 2……………………………….….……….6

6.Funciones………………………………………………………..….……...7

6.1.Función 1………………………………………….…….……..7

6.2.Función 2…………………………….………….……….…….8

7.Triggers………………………………………………………………….…...9

7.1.Trigger 1…………………………………………………….…..9

7.2.Trigger 2…………………………………………………….…..9

8.Acceso a sitio web…………………………………………………….10

9.Inicio…………………………………………………………………………....11

10.Empleado……………………………………………………………..…..12

11.Proveedor………………………………………………………………....13

12.Artículo……………………………………………………………………...14

13.Departamento…………………………………………………………..15

14.Tienda………………………………………………………………………..16

15.Ayuda…………………………………………………………………….…..17

16.Dificultades en el Desarrollo………………..………..……...18

17.Referencias…………………………………………………………..…..23

18.Conclusión y Comentarios……………………………………...23

Agradecimientos…………………………………………………...........24

Introducción

Nuestro proyecto es implementado en una tienda de abarrotes pequeña, lo que se realizó fue un inventario de todos los artículos en existencia para así poder organizar faltantes de productos y a su vez llevar un control de salidas y entradas en el sistema. Los artículos son registrados, esto para llevar un mejor control estos. En palabras simples, ayudaría a controlar a tener un mejor control y orden de los productos.

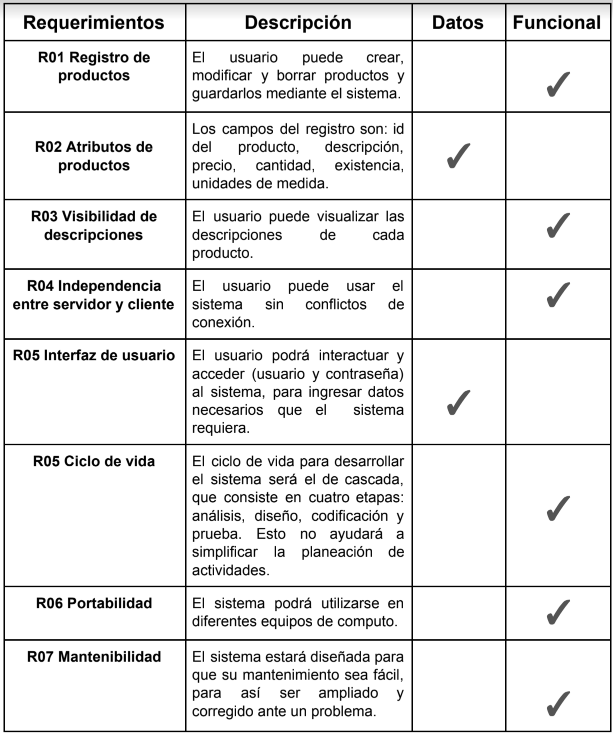
Descripción de la problemática

Como clientes de tiendas pequeñas (abarrotes), nos percatamos que la mayoría no cuentan con algún sistema informático que les permita registrar ventas, productos, existencias, etc, sus registros son manuales, escritos en papel, lo que puede ser tedioso, cansado y propenso a perder información.

Por resta razón, nosotros planteamos y desarrollamos un sistema para ser implementado en una tienda y así facilitar el manejo de información y trabajo.

Entre las ventajas que destacan al usar el sistema es:

* Mejorar el control de la tienda sobre sus productos.
* Detectar cuando y cuantos productos se necesitan para evitar mermas, perdida de dinero y espacio físico innecesario.
* Facilitar el trabajo en la tienda.

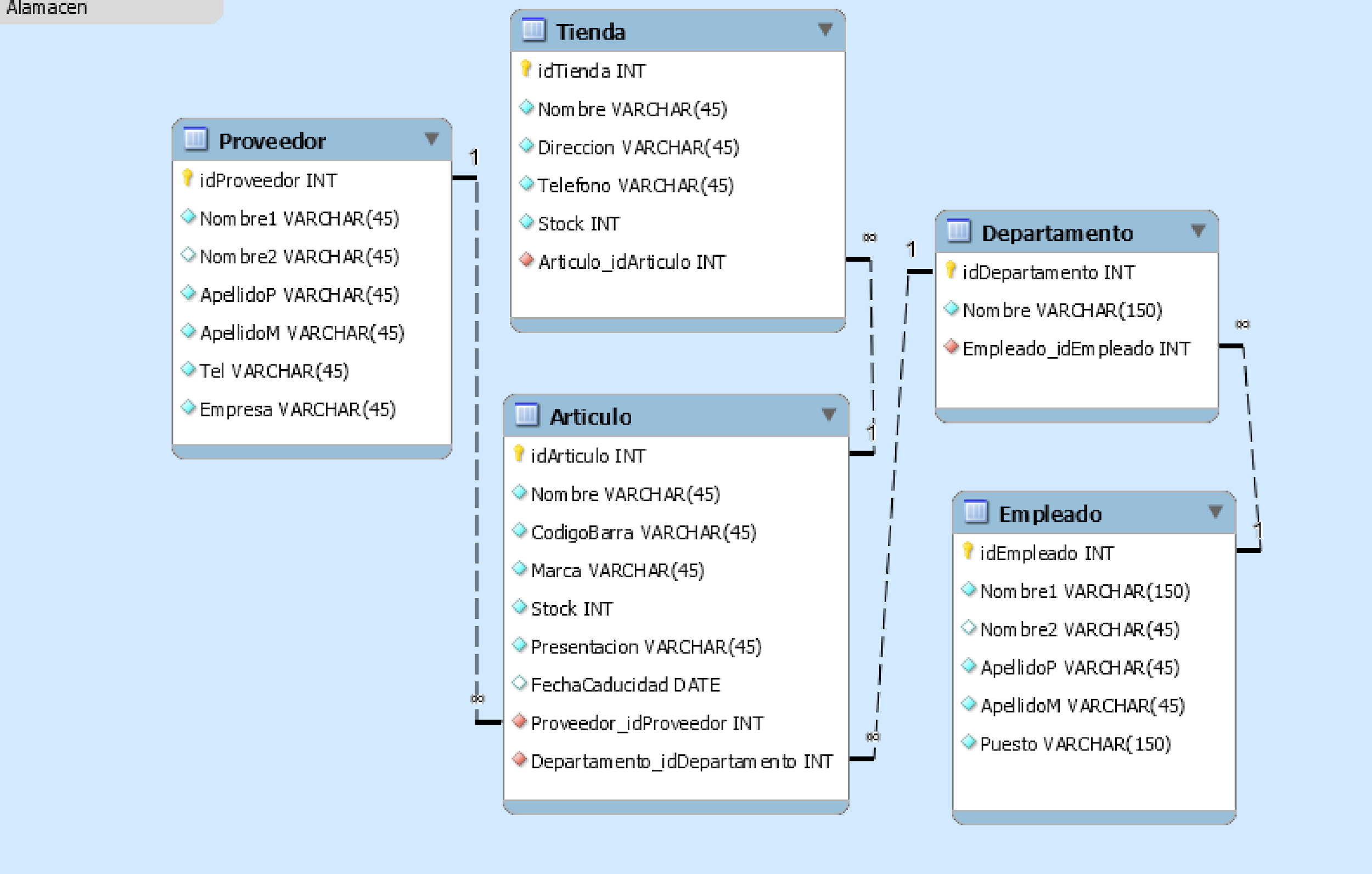
Requerimientos

Desarrollo e implementación

Modelo ER

La siguiente imagen contiene el modelo creado con la herramienta de software MySQL Workbench, que fue desarrollado e implementado en nuestro proyecto.

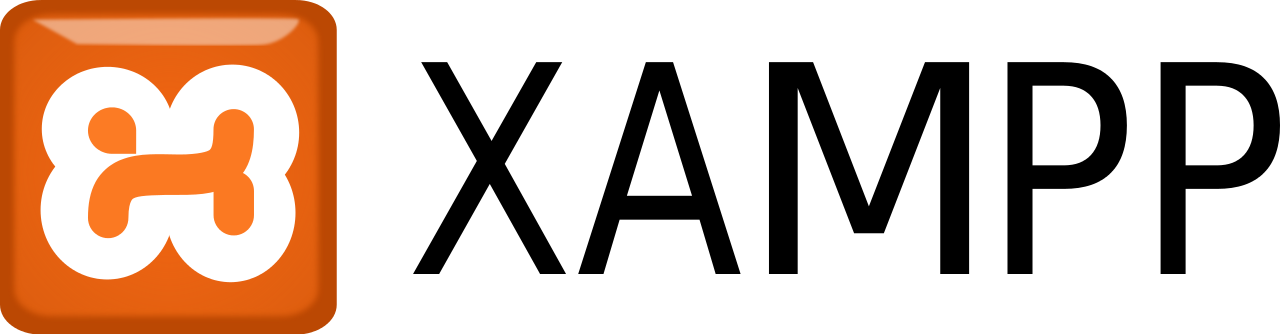
Cabe recalcar que se le aplicó formalización a cada una de las tablas para un mejor funcionamiento del sistema.



Una vez exportado el modelo, se procedió a importar la base de datos generada en un servidor local, utilizando el software XAMPP, de esta manera, a base de prueba y error se experimentó para determinar si sería la versión final. Seguido se procedió a buscar un sitio de alojamiento web para hospedar la base.

Primeramente se comenzó utilizando el sitio web de “infinitum free”, debido que no permite implementar funciones, procedimientos y triggers para la versión gratuita, recurrimos a otro sitio “000webhost” el cual fue utilizado para el proyecto, debido a que también es un servidor gratuito tiene limitaciones por lo que no pudimos mostrar la ejecución de las funciones.





Procedimientos almacenados

Procedimiento 1. Validación de Venta

En ocasiones no se tiene un registro exacto de la existencia de un producto, por lo que al realizar una venta o un pedido se debe satisfacer la demanda comercial. Por esta razón se creó este procedimiento para validar en el sistema que el total de salida de los productos sean menores a la existencia del producto y así saber cuándo, cuántos y cuáles productos se deben suplir para satisfacer la demanda y así evitar la pérdida de clientes.

*drop procedure if exists Venta*

*delimiter $$*

*create procedure Venta(in idTienda int, in Nombre varchar(150),*

*in Direccion varchar(150),in Telefono int,*

*in Stock int,in Articulo\_idArticulo int)*

*begin*

*declare existencia integer;*

*if (idTienda is not null) then*

*select Stock into existencia from articulo where idArticulo = Articulo\_idArticulo;*

*if (existencia > Stock) then*

*set existencia = Stock;*

*end if;*

*insert into tienda values(idTienda,Nombre,Direccion,Telefono,Stock,Articulo\_idArticulo);*

*end if;*

*end $$*

Procedimiento 2. Comprobación de un proveedor existente para agregar un artículo

Todos los artículos son surtidos por medio de proveedores, en el sistema se ha creado un procedimiento para comprobar que después de agregar un artículo exista un proveedor que haya surtido dicho producto, y de esta manera solo registrar artículos que pertenezcan a un proveedor registrado y así poder evitar registrar al sistema productos de dudosa procedencia, cabe mencionar que un proveedor puede surtir 1 o más productos.

*drop procedure Validar;*

*delimiter $*

*create procedure Validar(in id int,in Nom varchar(150), in Codigo varchar(150),in Mar varchar(150),*

*in sto int,in prese varchar(150),*

*in Fecha date,in idPro int,in idDepa int)*

*begin*

*declare idp int;*

*if idPro is not null then*

*select idProveedor into idp from proveedor where idProveedor = idPro;*

*insert into articulo values(id,Nom,Codigo,Mar,sto,prese,Fecha,idp,idDepa);*

*end if;*

*end $*

Funciones

Función 1. Existencia total de un producto especificado

Esta función fue hecha con el propósito de obtener un valor inmediato cuando se requiere saber la cantidad o existencia total de un producto en el inventario. Con este se evita el tener que contar físicamente pieza por pieza o producto, logrando optimizar las consultas.

Nota: para obtener la existencia actual se debe registrar correctamente la existencia previa para no obtener valores anómalos.

*drop function contarInventario;*

*DELIMITER //*

*create function contarInventario\_Articulo(Nom varchar(150)) RETURNS integer*

*BEGIN*

*declare suma integer;*

*select sum(stock) into suma from articulo where Nombre = Nom;*

*RETURN suma;*

*END //*

Función 2. Existencia total de un producto especificado

Buscar empleados en el sistema puede llegar a ser una labor tediosa, por lo que se creó un función para buscar por nombre a un empleado.

La función consiste en solo insertar en el campo especificado el nombre del empleado, la función devuelve cuántos empleados tienen el mismo nombre (caracteres similares) y de esta forma evitar pérdida de recursos y/o tiempo en la búsquedas manuales.

*use basefinal;*

*drop function buscarEmpleado;*

*DELIMITER //*

*create function buscarEmpleado (buscaNombre varchar(20)) RETURNS integer*

*BEGIN*

*declare resul integer; #varchar(150);*

*select count(empleado.Nombre1) into resul from empleado where Nombre1 = buscaNombre;*

*return resul;*

*END //*

Triggers

Trigger 1. Actualizar stock (existencia) de un artículo

Este disparador funciona de la mano con el procedimiento “Venta” ya que después de verificar la existencia de un artículo para poder salir de la tienda, se actualiza el stock con el disparador y así reflejar la existencia actual del producto.

*drop trigger actualiza\_articulo*

*delimiter $*

*create trigger actualiza\_articulo after insert on tienda*

*for each row*

*begin*

*update articulo set Stock = Stock - new.Stock where idArticulo = new.Articulo\_idArticulo;*

*end $*

Trigger 2. Inserción de fecha

Todos los productos ingresados al sistema deben de contar con una fecha de caducidad, por lo que se ha implementado este disparador para agregar una fecha, esta es un mes de diferencia mayor a la actual. Es un estándar de seguridad que se ha implementado para verificar que los productos que están por expirar en el mes siguiente (tomando en cuenta la fecha actual de ese momento) deben revisarse y notificar que la fecha no fue ingresada en el momento de su captura en el sistema de inventario.

*delimiter $*

*CREATE TRIGGER after\_articulo\_insert*

*before insert on articulo*

*for each row*

*begin*

*declare aux date;*

*if(new.FechaCaducidad is null) then*

*select DATE(DATE\_ADD(now(), INTERVAL 1 MONTH)) into aux;*

*set new.Fechacaducidad = aux;*

*end if;*

*end $*

Acceso a sitio web

Este es el enlace que direcciona a la página web para su revisión.

# <https://securable-diseases.000webhostapp.com/>

# 

# Inicio

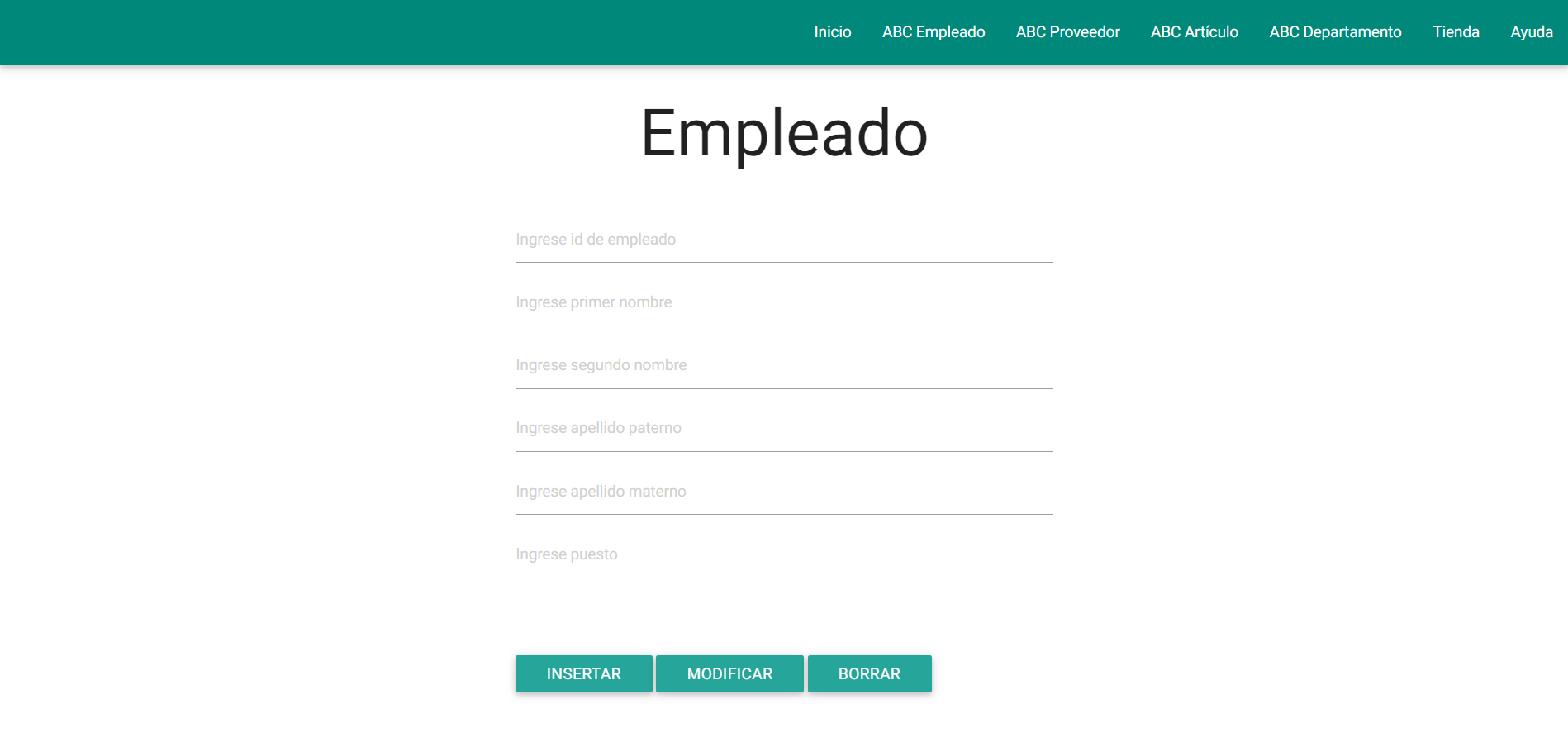
Una vez ingresado al sitio, será direccionado a la página de principal. En la parte superior se incorporó un menú agradable a la vista para el fácil acceso de las pestañas como lo muestra la imagen.

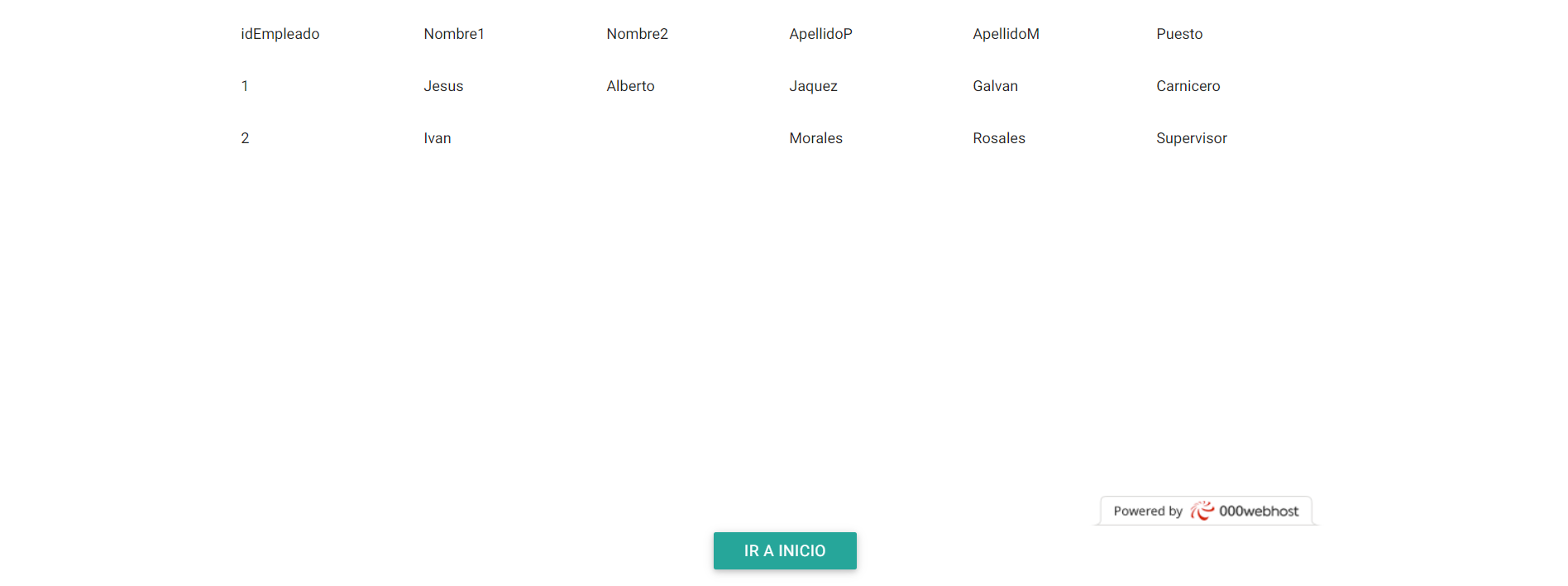


# 

# Empleado

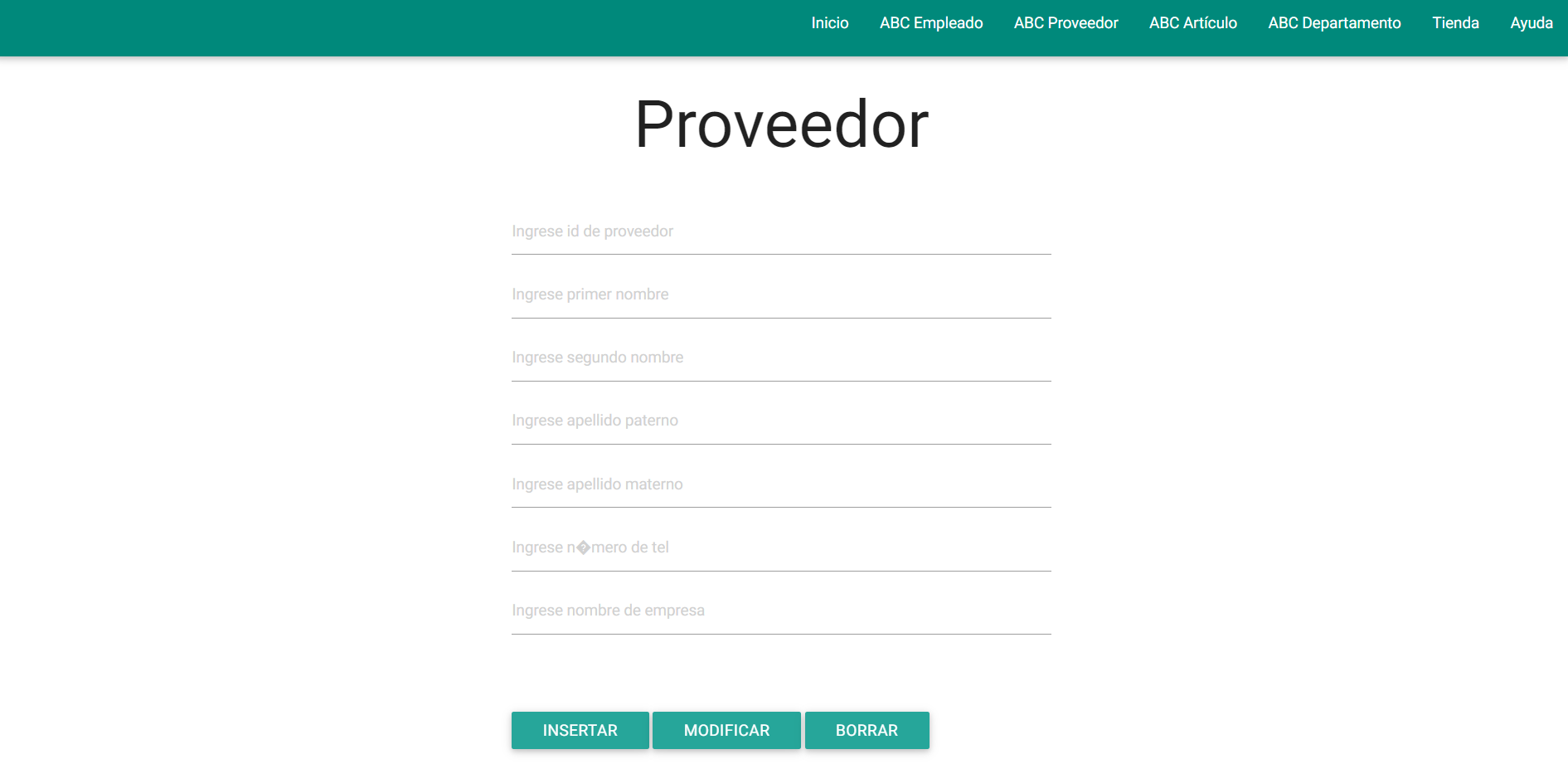
En esta pestaña se encuentra un formulario a completar para registrar empleados que pertenecen a un departamento, la tabla muestra los registros y modificaciones realizadas, el campo “ingresa idEmpleado” es un valor único, además se habilitó que el “ingresa segundo nombre” fuera nulo ya que no muchas personas tienen este nombre. El campo “ingresa puesto” es donde se ingresa el puesto al que pertenece el empleado. Todo los campos están validados.

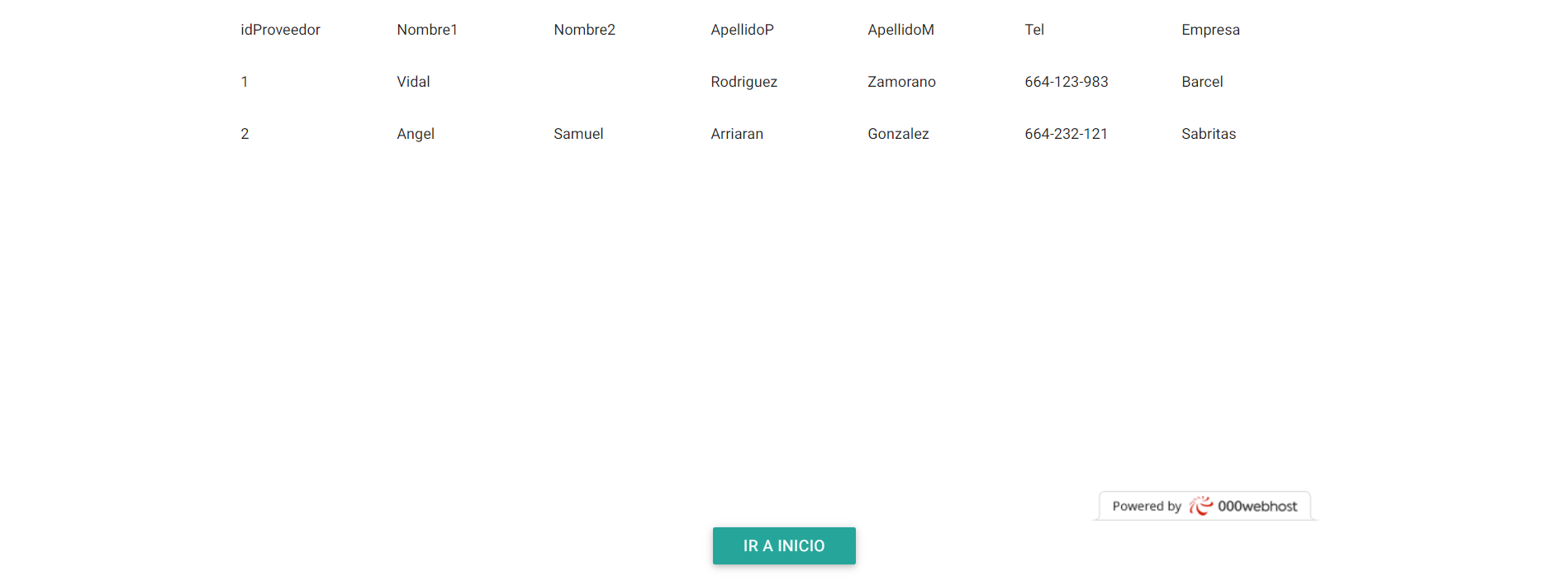




# Proveedor

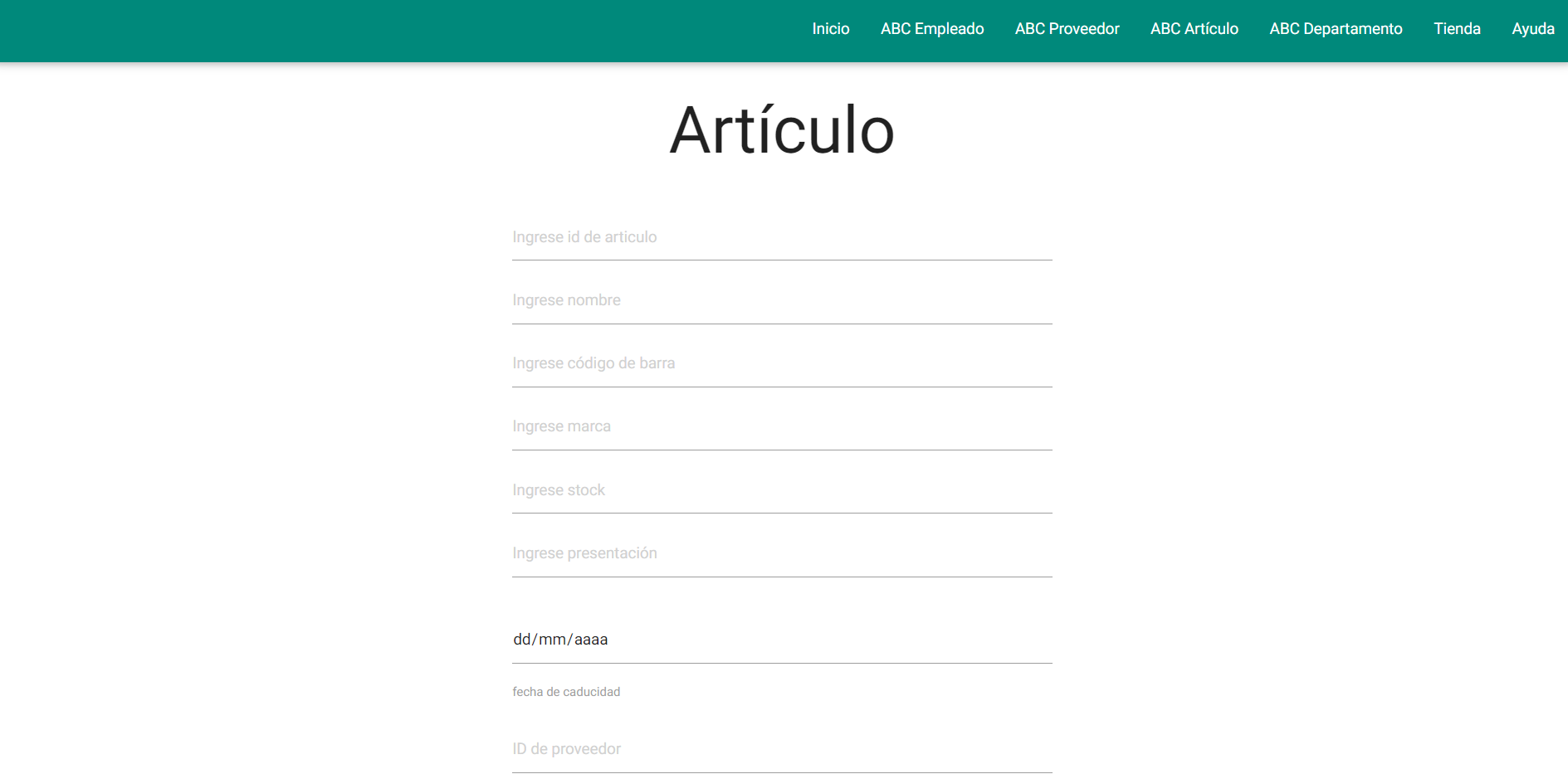
La pestaña cuenta con un formulario para ingresar los datos de los proveedores. También cuenta con las mismas validaciones que la pestaña empleado a excepción de que se agregó un campo para teléfono y empresa proveniente del proveedor.

****



# Artículo

La pestaña muestra un formulario para registrar los artículos que entrarán a el almacén, cada uno tiene un identificador único en caso de que se repita el código de barras, stock para ingresar la cantidad de productos y para revisar la existencia de estos. Todo producto debe contar con una fecha de caducidad e identificador de proveedor y departamento para saber a cual corresponde. Aquí se emplea el procedimiento de validar a el proveedor.





# Departamento

El apartado muestra un formulario con los datos básicos del departamento. Se agregó el campo “ingrese id de empleado” para tener un control de que empleados pertenecen a un departamento específico.

# 

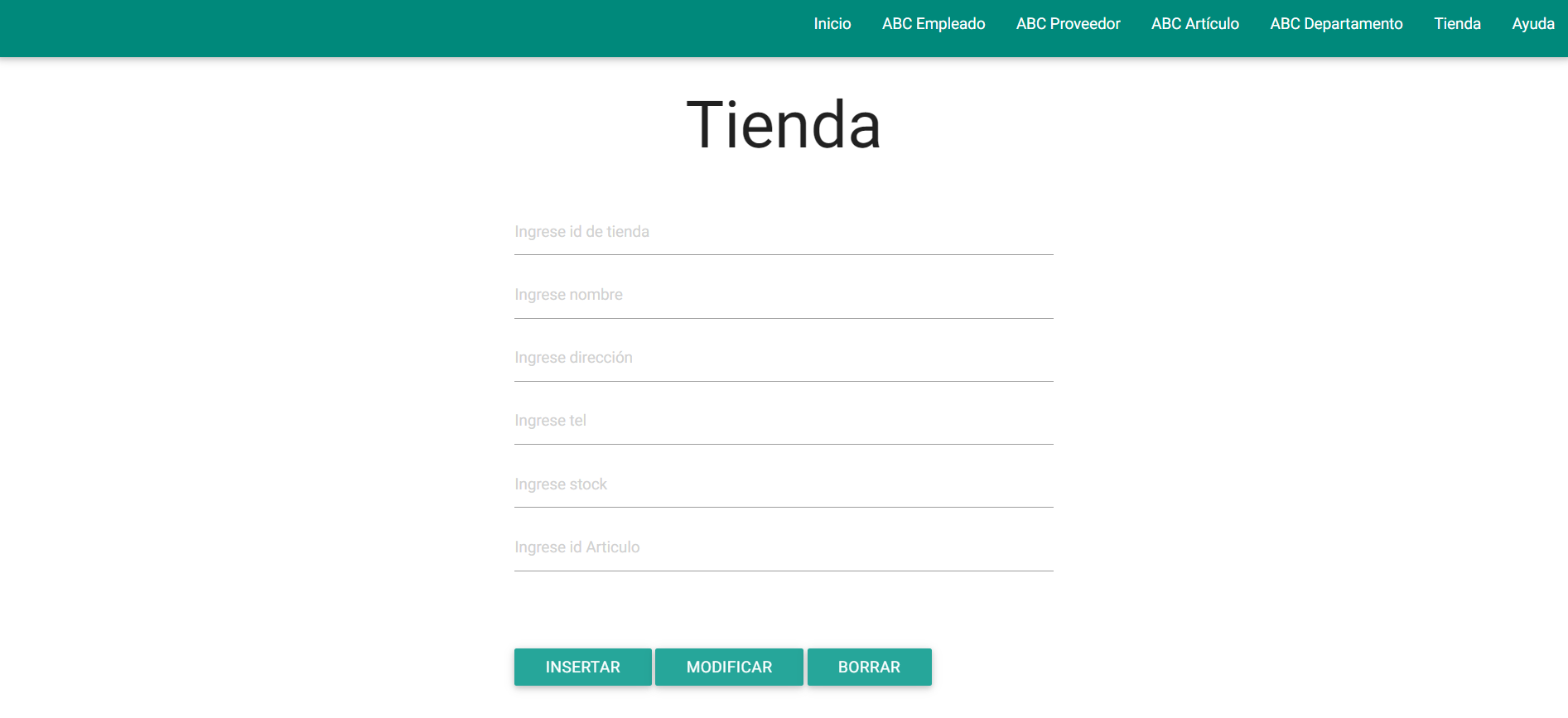
# 

# 

# 

# Tienda

Esta pestaña también cuenta con un formulario a llenar, se le agregó el campo de “ingresar id de artículo” para saber que productos pertenecen a una tienda identificada por su id único. Se incorporó un disparador para restar la existencia de artículos que es activado cuando el procedimiento realiza los cálculos para verificar que se cuenta con los productos necesarios para llevar a cabo el proceso que es reflejado en la pestaña de artículos.





Ayuda

En esta sección se explica como insertar, modificar y borrar los datos.

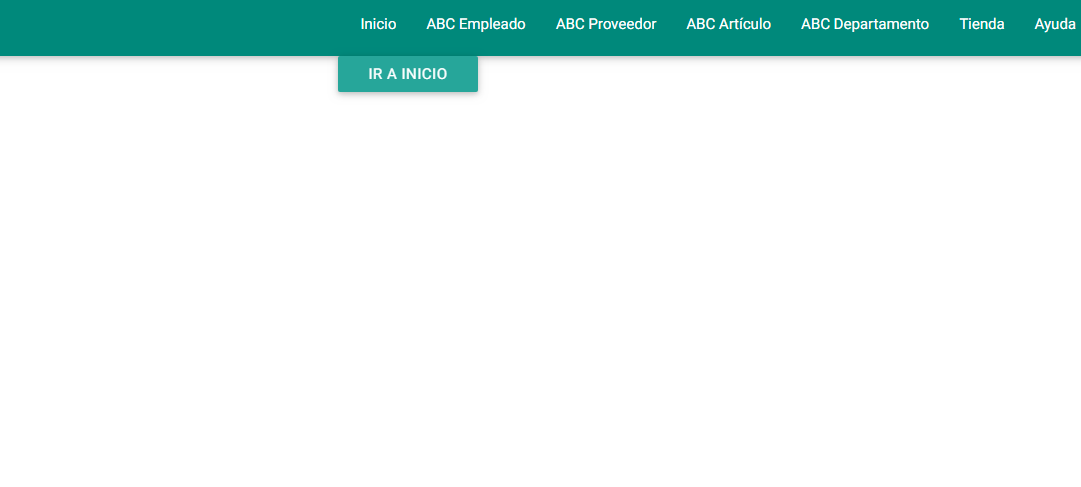
Nota: Este sistema fue diseñado para uso de personal autorizado para evitar la pérdida de información.



# Dificultades en el desarrollo

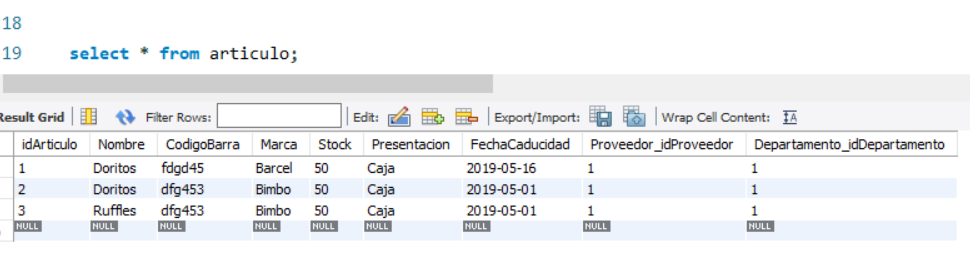
En la implementación de las funciones a nuestro sistema, tuvimos unas complicaciones que fueron el mostrar el resultado gráficamente, previo a esto, fueron probadas y validadas localmente, por lo que se agrega evidencia de su correcto funcionamiento.

El error generado era el siguiente. Por lo que se decidió remover la llamada a las funciones del sistema para el funcionamiento de los otros procesos.

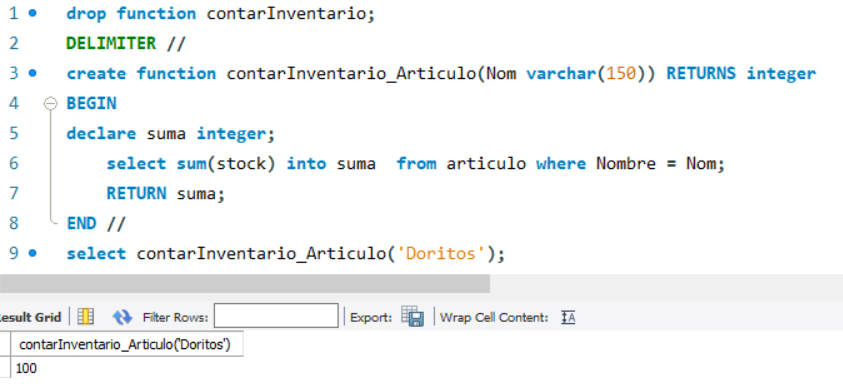


1. Esta función suma toda la existencia del artículo especificado para obtener el total de estos que se encuentran en el almacén.

La imagen muestra que el artículo “Doritos” tiene una existencia de 50 cajas surtidas por el proveedor “Bimbo” y 50 cajas surtidas por el proveedor “Barcel”.

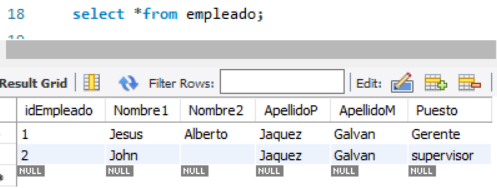
****

La siguiente imagen refleja que después de mandar llamar a la función con el parámetro a buscar (Doritos), retorno la existencia de este en el almacén (100 cajas).

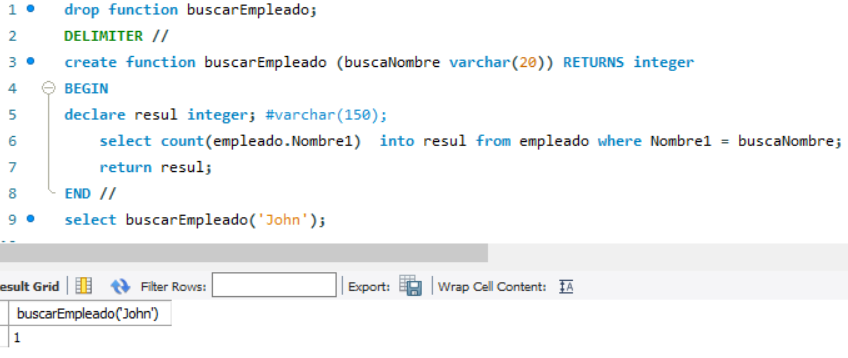


2. Esta función devuelve la cantidad de empleados con el mismo nombre.

La imagen muestra que existe solo 1 empleado con el nombre “John”.



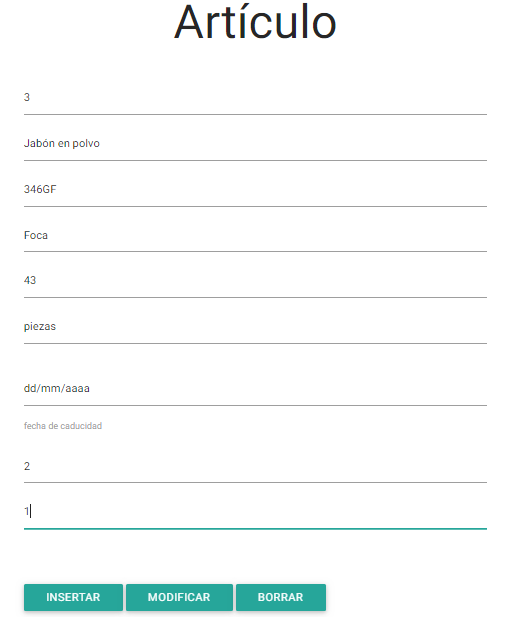
Al aplicar la función se devuelve el resultado, solo existe 1 empleado con el nombre de “John”.



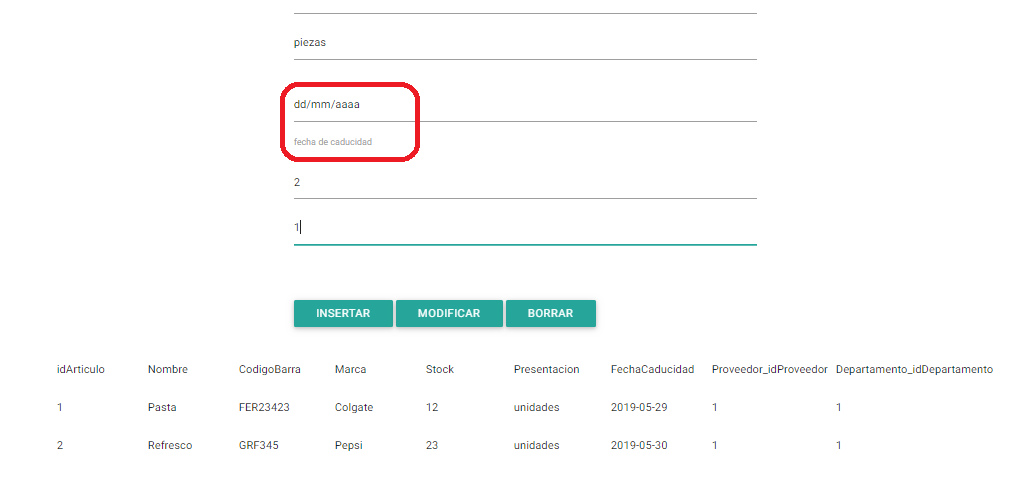
# 

3. Lo siguiente es el trigger para agregar una fecha a los artículos que no se les agrego fecha de caducidad. Este consiste en insertar la fecha del mes siguiente del presente año en el que se registra el artículo, al percatarse de que los productos a expirar en el mes siguiente fue por que no se les ingreso fecha de caducidad.

La imagen muestra el registro de un artículo sin agregar fecha par ver la ejecución del trigger en la tabla de datos.



Al insertar el artículo sin fecha, se puede apreciar que el trigger ha sido disparado correctamente, mostrando la fecha del mes siguiente al que se registró (fecha de registro: 30/05/2019, fecha agregada: 30/06/2019 )





# 

# Referencias

Información de las presentaciones de clase en classroom con la profesora:

Sukey Sayonara Nakasima Lopez.

<https://classroom.google.com/u/1/c/Mjc4MzY0ODU2NzZa>

# MySQL 8.0 Reference Manual. <https://dev.mysql.com/doc/refman/8.0/en/>

Plantilla. <https://www.templatemonster.com/es/plantillas-web-tipo/>

# Conclusiones y comentarios

# En el desarrollo de este proyecto pudimos aprender y aplicar conocimientos adquirido en clases y laboratorio sobre el lenguaje Mysql, php y html. Debido a que son los lenguajes básicos (indispensables) para poder llevar a cabo nuestra página web con la base de datos conectada.

También aprendimos a como ver un problema del exterior empresarial (de la vida real) y aplicar una solución real empleando una base de datos, de igual manera se aprendío a cómo se administra un servicio de host web para que desde una liga se pueda acceder y hacer uso de nuestro proyecto realizado.

Ante no poder mostrar la ejecución de las funciones en el sistema, realizamos consultas en internet y con la profesora Dra. Olivia Mendoza Duarte sobre los problemas con las funciones, concluyendo que el sistema operativo permite que se ejecuten algunas instrucciones sin mostrar error, sin embargo los servidores online que utilizan Linux, son más estrictos. Intentamos solucionar el problema, sin embargo como evidencia agregamos las capturas del funcionamiento de estas usando el sistema local con la misma base que se cargó al sitio host web.

Agradecimientos

De antemano, agradecemos el apoyo y la paciencia de los docentes Olivia Mendoza Duarte y Sukey Sayonara Nakasima Lopez por ser nuestros profesores a lo largo del semestre 2019-1, les deseamos que sigan llenando sus vidas de éxito. Muchas gracias.