***UNIVERSIDAD MARIANO GALVEZ DE GUATEMALA***

***MAZATENANGO, SUCHITEPÉQUEZ***

***Ingeniería en Sistemas***

***Ingeniero Axel Aguilar***

***Desarrollo Web***

**Fase I**

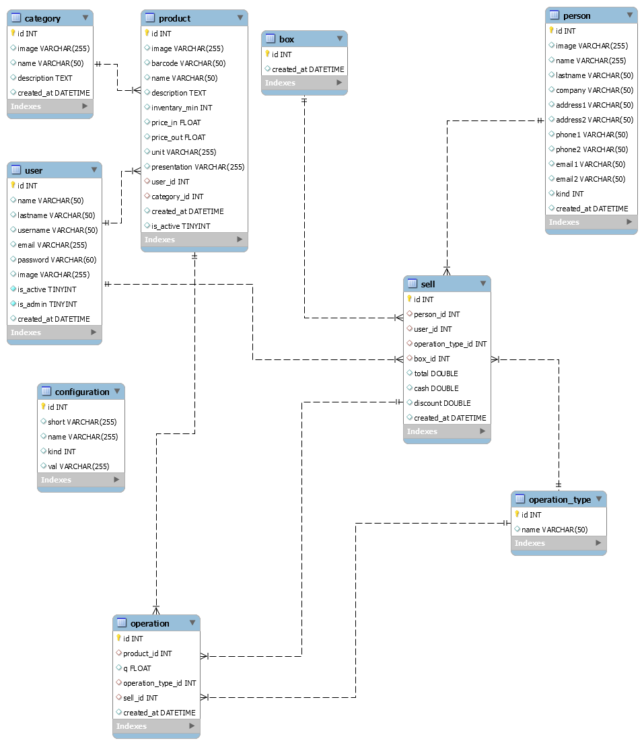
**Proyecto #2 Inventario**

**Daniel Omar Alecio de León**

**3090-14-7861**

**25-08-2018**

**Modelo de base de datos**



La base de datos del sistema de inventario está conformada por 9 tablas las cuales son:

* box
* category
* configuration
* operation
* operation\_type
* person
* product
* sell
* user

**Category**

La tabla category sirve para las categorías de los productos del sistema.

Campos

id: Id auto incremental

image: Imagen de la categoría, no implementado

name: Nombre de la categoría, el que se muestra en el sistema

description: Descripción de la categoría

created\_at: Fecha de creación

En la tabla product existe un campo llamado category\_id que se relaciona con el id de una categoría.

**Product**

La tabla product es la tabla principal de la base de datos, la tabla product simboliza los productos y cuenta con los siguientes campos:

id: El id o llave primaria del producto

image: se usa para almacenar el nombre de la imagen del producto

barcode: el codigo de barras

name: el nombre del producto

description: una breve descripción del producto

inventary\_min: almacena el valor de cantidad mínima en inventario

price\_in: el precio de entrada, precio de compra o precio de abastecimiento

price\_out: precio de salida o precio de venta

presentation: presentación en bolsa, en botella en caja, etc …

unit: unidad, metros, kilos, libras, onzas, etc.

user\_id: almacena el id del usuario que da de alta el producto

category\_id: id de la categoría a la cual pertenece el producto

is\_active: el producto esta activo o inactivo

created\_at: fecha de creacion del producto

Consultas útiles:

Mostrar todos los productos: select \* from product

Mostrar productos por category\_id: select \* from product where category\_id=1

Mostrar productos activos: select \* from product where is\_active=1

Mostrar productos creados por cierto usuario: select \* from product where user\_id=1

**Person**

La tabla person simboliza un tipo de persona dentro del sistema, las “personas” por default son: clientes y proveedores definidas por el campo kind, los campos son:

id: id o llave primaria

image: imagen o foto de la persona

name: nombre completo de la persona

lastname: apellidos de la persona

company: nombre de la empresa

address1: dirección o domicilio

address2: dirección o domicilio

phone1: teléfono

phone2: teléfono

email1: correo electrónico

email2: correo electrónico

kind: tipo de persona, 1=cliente, 2=proveedor

created\_at: fecha de creación

Consultas útiles

Mostrar clientes: select \* from person where kind=1

Mostrar proveedores: select \* from person where kind=2

**Box**

Box sirve para relacionar los cortes de caja.

Campos

id: Id autoincremental

created\_at: Fecha de creacion

En la tabla sell existe un campo llamado box\_id, entonces cuando se crea una entrada en la tabla box, se asigna el id de la entrada en la tabla box al campo box\_id de las “ventas” relacionas.

**Operation Type**

La tabla operation\_type es una biblioteca que contiene los tipos de operación del sistema.

Campos

id: Id autoincremental

name: Nombre del tipo de operación

Esta tabla normalmente va de la mano con la tabla operation.

Los valores de la tabla operation\_type estan definidos por default y no se pueden cambiar, en caso de que se cambien pueden afectar el funcionamiento normal del sistema:

1= entrada

2= salida

**Sell**

La tabla sell es una tabla que agrupara las operaciones, esta relacionado con las ventas o abastecimientos según el operation\_type\_id los campos de la tabla sell son:

id: el id de la venta o abastecimiento

person\_id: el id del cliente para ventas o proveedor en abastecimientos

user\_id: el id del usuario que realiza la venta o abastecimiento

operation\_type\_id: el tipo de operacion, entrada o salida

box\_id: el id de caja, siempre es NULL hasta que se crea un corte de caja se le asigna el id del nuevo corte de caja

total: el total de la venta o abastecimiento

cash: el total del pago realizado por el cliente

discount: el descuento en caso de que se aplique

created\_at: la fecha de creacion de la entrada

Consultas útiles

Mostrar ventas: select \* from sell where operation\_type\_id=2

Mostrar abastecimientos: select \* from sell where operation\_type\_id=1

Mostrar ventas por corte de caja: select \* from sell where operation\_type\_id=2 and box\_id=1

Mostrar ventas por rango de fecha INICIO-FIN (Formato “YYYY-MM-DD”): select \* from sell where operation\_type\_id=2 and (date(created\_at)>”INICIO” and date(created\_at)<=”FIN”)

**Operation**

La tabla operation cuenta con los siguientes campos:

id: el id de operación, la llave primaria

product\_id: el id del producto de la operación

q: la cantidad de producto

operation\_type\_id: el tipo de operación, entrada o salida

sell\_id: el id de la venta o abastecimiento

created\_at: la fecha de creación

Consultas SQL

Sumar total de entrada o abastecimientos por producto: select sum(q) from operation where operation\_type\_id=1 and product\_id=1;

Sumar total de salidas o ventas por producto: select sum(q) from operation where operation\_type\_id=2 and product\_id=1;

**User**

La tabla user sirve para almacenar los usuarios que tienen acceso al sistema, los campos son:

id: id o llave primaria

name: nombre del usuario

lastname: apellido

username: nombre de usuario

email: email

password: contraseña de acceso

image: imagen o foto de perfil

is\_active: valor booleano para identificar si esta activo o inactivo

is\_admin: valor booleano para identificar si el usuario es administrador

created\_at: fecha de creación del usuario

Consultas útiles

Login valido del usuario: select \* from user where (email=”EMAIL” and password=”PASSWORD”) and is\_active=1

Configuration

La tabla configuration es una tabla experimental que sirve para usar configuraciones o ajustes generales que afectan al funcionamiento del sistema.

Campos

id: Id autoincremental

short: Nombre corto, normalmente se utiliza como una llave para acceder a los valores de las configuraciones en lugar de usar el id

name: Es el nombre de la configuración, este se muestra en el sistema.

kind: Tipo de valor puede ser textual, numérico, si/no, imagen. se usa un valor numérico, por ejemplo, tipo textual=1, numérico=2…

val: Valor almacenado, es el valor referente a la configuración

**Servidor a utilizar**

**IIS**

Internet Información Services, es un servidor web y un conjunto de servicios para el sistema operativo de Microsoft Windows

Ventajas:

Es confiable, seguro y fácil de utilizar

No tiene coste adicional, ya que viene incluido en el paquete Windows

Desarrollada por Microsoft, dándole un respaldo importante

Se puede administrar a través de Internet

Soporte ODBC integrado

**Soporta .NET framework y lenguaje ASPX.**

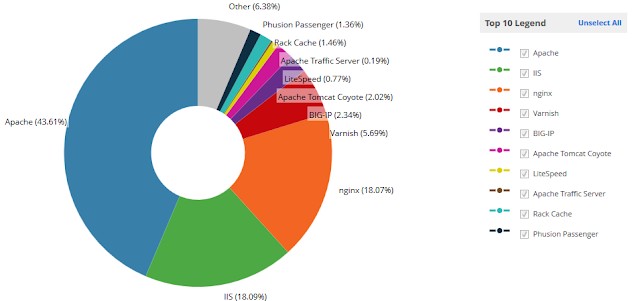
Desventajas:

Su licencia no es gratuita

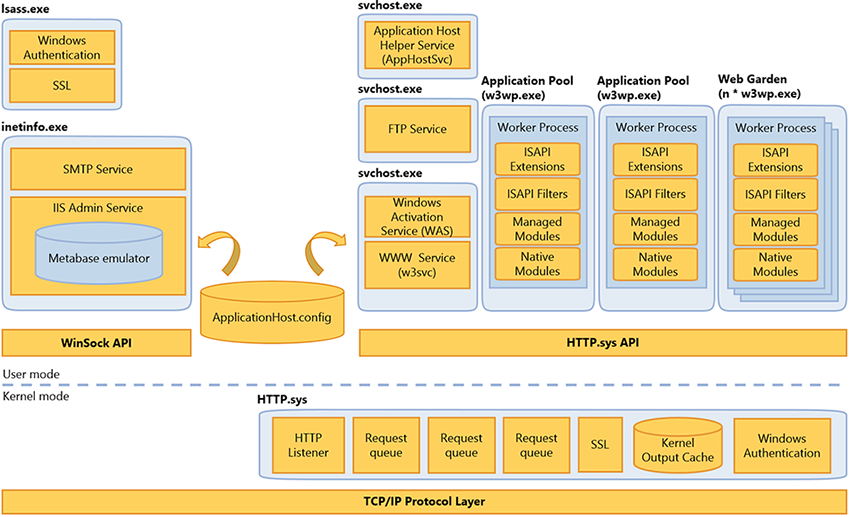
No tiene multiplataforma

Resulta difícil de controlar la dirección

Código fuente propietario



**Arquitectura**



**Recomendaciones de la arquitectura del servidor**

En términos de disponibilidad en páginas web, facilidad de programación, flexibilidad, la recomendación es el uso de Apache, aunque este servidor tiene acceso a ASPX, con PHP se puede suplir casi cualquier necesidad que nuestra web necesite.

**Como lograr alta disponibilidad en el servidor**

Se puede lograr al utilizar Network Load Balancing (NLB) en donde se tendrá a los dos servidores activo - activo. Si uno de los nodos cae, el que está operativo continúa con la carga.

Network Load Balancing: Un balanceador de carga actúa como único punto de contacto para los clientes. El balanceador de carga distribuye el tráfico entrante entre varios destinos, como instancias Amazon EC2. Esto aumenta la disponibilidad de la aplicación. Puede agregar uno o varios agentes de escucha al balanceador de carga.