

Proyecto: DEPSYS

Documentación del proyecto en general

Michael Villalba Sotelo

DEPSYS

"SISTEMA DE DEPOSITO"

El siguiente proyecto surge del planteamiento de una necesidad, mantener una organización en la información. Este planteamiento no se necesita usar necesariamente dentro de una empresa con activos y la creciente información que puede llegar a tener, información que muchas veces se necesita organizar dentro de una hoja de calculo sin ningún tipo de orden en si. O en algún tipo de documento que bien puede ser borrado con facilidad; o también en algún medio de almacenamiento físico que puede ser tedioso ocupando mucho espacio, y que al realizar una búsqueda no se pueda encontrar el documento o la información deseada debido a la baja cantidad de retención que tiene el cerebro humano.

Para esto se plantea un sistema, que maneje la información de una forma generalizada sin un proposito conciso, sino adaptado a las necesidades del usuario, manteniendo un desarrollo separado de las partes que intervienen dentro de un software: Datos, Procedimientos e Interfaz de usuario.

El planteamiento se mantiene sencillo, tratar de manejar lo más separados posibles las 3 partes. Los datos que sin ningún tratamiento no son útiles, la programación que sin datos no realizara nada y la forma de interactuar con el usuario que si no fuera utilizado por el no tendría sentido. Las tres partes fundamentales pero que entre más separadas sean más dinamicas pueden ser.

Se empezara por la estructura de datos, aquella que será utilizada por la programación.

Estructura de Datos

Se utilizará una base de datos mariadb, implementación libre del gestor de bases MYSQL. Contiene todas las herramientas esenciales así como buena documentación y libreria para poder gestinarla desde el lenguaje de programación C.

Hasta el momento la estructura de datos se divide en:

–Datos Generales: Que son todos aquellos datos ingresados por el usuario para su gestión, como consulta, alta y baja de registros y aquellos que el programa utilizara para darles una funcionalidad.

–Datos del programa: Son los datos que pueda usar el programa internamente para su funcionamiento.

ESTRUCTURA DE DATOS GENERALES

Los diferentes elementos a tomar en cuenta para la esctructuración de datos son:

Todos los recursos dependen de la campaña en la que se implementan por esto el elemento principal del sistema es: "Campaña".

Los recursos implementados dependen del servicio contratado para el cliente, debido a esto el elemento secundario es: "Servicio".

Un servicio es constituido por multiples sistemas, el elemento terciario es: "Sistema".

Un sistema se construye con diferentes herramientas, el cuarto elemento es: "Herramienta".

Elemento Principal "Campaña"

Una campaña tiene:

- *Nombre
- *Identificador
- *Condición de Activa o Inactiva

Elemento Secundario "Servicio"

Un servicio tiene:

- *Nombre
- *Identificador
- *Relación con la campaña
- *Descripción

Elemento Terciario "Sistema"

Un sistema tiene:

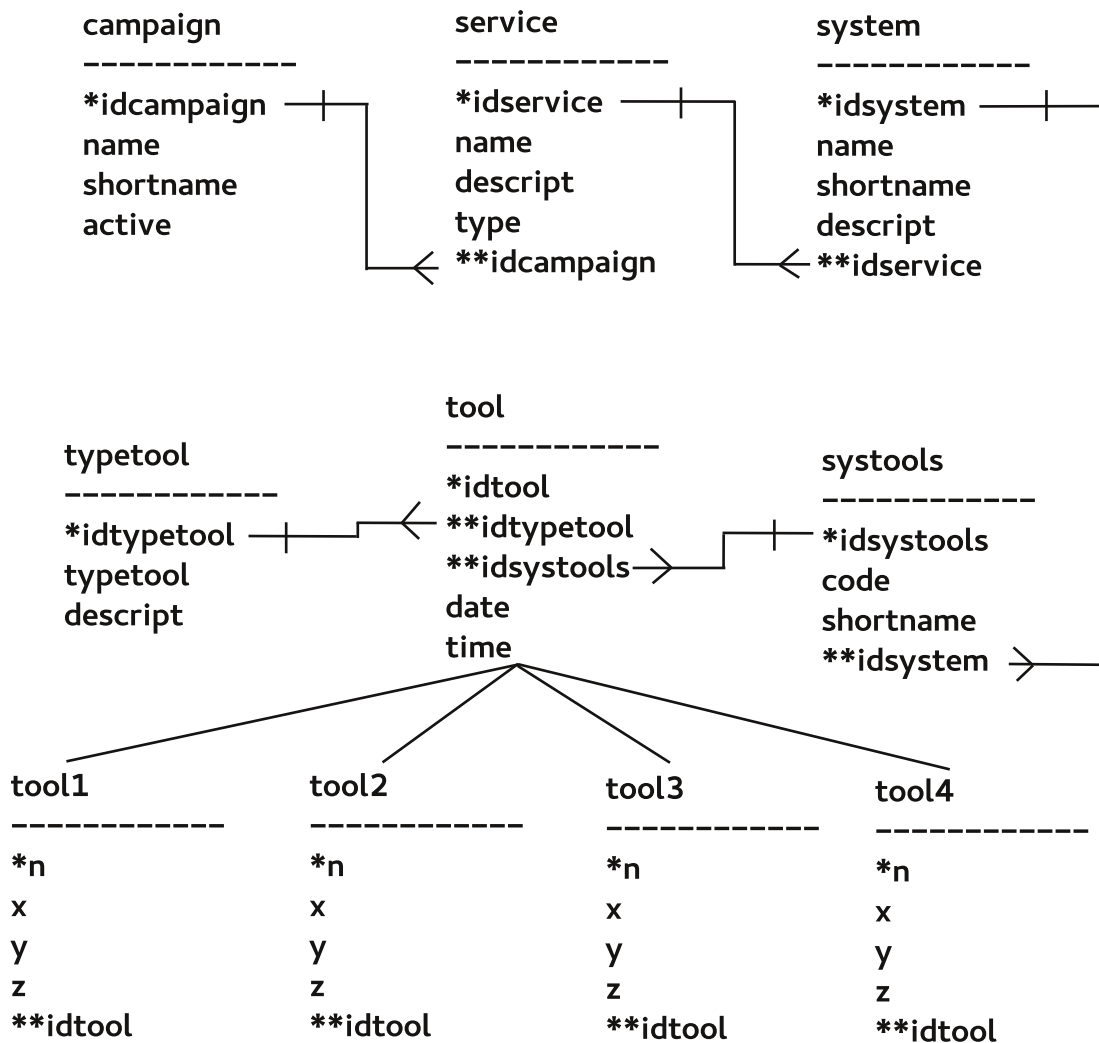
- *Nombre
- *Identificador
- *Una relación con el servicio

Elemento Cuatro "Herramientas"

Una Herramienta Tiene:

- *Nombre
- *Una clave
- *Una abreviación
- *Una relación con el sistema

MODELO ENTIDAD-RELACIÓN



DICCIONARIO DE DATOS

campaing

Esta tabla contiene la información del elemento "Campaña", gracias a una campaña se crean los diferentes servicios, la tabla contiene los siguientes campo:

idcampaign: Se refiere al identificador del registro, es un tipo de campo INT.

name: Nombre que representa a la campaña, tiene tipo de campo VARCHAR(30).

shortname: Abreviación del nombre de la campaña. Tipo de campo VARCHAR(20).

active: Condición para saber si una campaña esta activa o inactiva, tipo de campo BINARY.

service

Esta tabla contiene la información del elemento "Servicio", el servicio viene relacionado con el sistema, la tabla contiene los siguientes campos:

idservice: Se refiere al identificador del registro, es un tipo de campo INT.
name: Nombre que representa al servicio, tiene tipo de campo VARCHAR(30).
descript: Campo donde se describe el servicio. Tipo de campo VARCHAR(200).
type: Tipo de servicio, es un tipo de campo VARCHAR(30).
idcampaign: Campo para relacionar el servicio con la campaña, INT.

system

Esta tabla contiene la información del elemento "Sistema", el sistema contiene las herramientas usadas para brindarle el servicio al cliente, los campos que contienen son:

idsystem: Se refiere al identificador del registro, es un tipo de campo INT.
name: Nombre del sistema, tiene tipo de campo VARCHAR(30).
shortname: Abreviación del nombre del sistema. Tipo de campo VARCHAR(20).
descript: Descripción breve del sistema, este es un campo VARCHAR(200).
idservice: Relación con la tabla de servicio, es un campo tipo INT.

systool

Esta tabla contiene la información del elemento "Herramientas", las herramientas del sistema tienen una relación con el sistema y otra con la herramienta, la tabla contiene los siguientes campos:

idsystools: Se refiere al identificador del registro, es un tipo de campo INT.
code: Un la clave con la que se identifica a la herramienta, VARCHAR(20).
shortname: Nombre de la herramienta del sistema, VARCHAR(30).
idsystem: Relación con la tabla de sistema, INT.

typetool

Esta tabla contiene los tipos de herramientas, sirve como control para crear una nueva herramienta, la tabla contiene los siguientes campos:

idtypetool: Identificador del registro, es un tipo de campo INT.
typetool: Contiene el tipo de herramienta, es un VARCHAR(20)
descript: Descripción del tipo de herramienta, VARCHAR(200).

tool

Esta tabla sirve como tabla relacional para gestionar la creación y consulta de una herramienta, sus campos son:

idtool: Se refiere al identificador del registro, es un tipo de campo INT.

idtypetool: Es un identificar de relación con la tabla tipos de herramienta, INT.

idsystools: Relación con las herramientas del sistema, INT.

date: fecha de creación de la herramienta, tipo de campo DATE.

time: Hora de creación de la herramienta, tipo de campo TIME.

tooln

Las diferentes tablas de las herramientas, cada herramienta tiene sus diferentes cualidades por lo que se requiere una tabla para cada herramienta, debe tener los siguientes campos:

idtooln: El identificador del registro, tipo INT.

propietyn: Las diferentes propiedades de la herramienta.

ESTRUCURA DE DATOS PARA APLICACIÓN

Menú:

El programa pensado realizara varias funciones para el objetivo diseñado, consultar, insertar y borrar información dentro de una base de datos. Pero no se planea hacer un sistema estatico, sino que este puede expandirse para hacerlo de la forma más modular posible sin necesidad de recompilar el código las opciones del menú a las cuales se pueden ingresar para realizar las diferentes funciones se guardaran en una tabla llamada Menú.

ALGORITMOS DEL SISTEMA

Seguido de la estructuración de los se detallaran los procedimientos a seguir para su manejo.

Un deposito es un "lugar" donde se colocan "cosas", en el sistema este lugar es una base de datos, pero no solo se depositaran las "cosas" (información) sino que se obtendra y consultara despues de ser ingresada.

Para depositar la información es necesario:

- Conocer la Información a depositar.
- Recoger la selección del usuario para mostrarle los campos a llenar.
- Verificar que lo que este llenando el usuario sea del tipo de campo correcto.
- Recoger los campos que lleno el usuario.
- Verificar que la información obtenida sea correcta.
- Sí la información es correcta guardar la información

En caso de no conocer la información el sistema puede guiar al usuario para ingresarla:

- Preguntar el tipo de información que desea ingresar.
- ¿Qué información deseas ingresar?
- Mostrarle las diferentes opciones de información para ingresar.
- Cargar los tipos de información y darle a elegir las diferentes opciones.
- Recoger la selección del usuario.
- Dependiendo del tipo de información seleccionada mostrar los campos a llenar.
- Cargar los campos a llenar y mostrarlos al usuario para que los llene.
- Guiar al usuario para el llenado de los campos.
- Cargar los datos llenados.
- Guardarlos.