

PSP0 Etapas

Héctor Isaí De León Sevilla



9 de febrero de 2017

UTRM

Prof. Christian Bernal

# Planeación.

Objetivo

Crear una aplicación donde se registre el cliente y se pueda asignar una categoría dependiendo el área en donde se va a trabajar. La cual involucra las tecnologías de la información como motor para satisfacer las necesidades del cliente reduciendo costos a lo largo del tiempo y aumentando la productividad.

Es necesario el diseño de una estrategia que permita utilizar la tecnología para la identificación de las circunstancias que afectan primordialmente la relación con nuestros clientes, analizar la situación y decidir de manera oportuna coordinada por todas las áreas de contacto y soporte.

Alcance

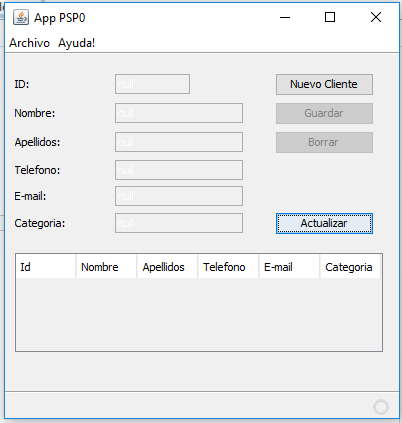
El sistema se desarrollará en su primera etapa realizando las funciones de agregar, modificar y eliminar un cliente, así como una categoría.

Este sistema deberá cumplir con requerimientos específicos demandados por el cliente.

Requerimientos

* No poder eliminar una categoría, si está relacionada con un cliente.
* Nombre y apellidos son obligatorios.
* Correo electrónico valido.
* El nombre de la categoría no puede quedar vacío.
* El correo electrónico no se puede duplicar.

**Diseño.**



**Codificación.**

El lenguaje de programación en el que se desarrolla la app es java, a continuación incluyo un fragmento del estándar decodificación que se utiliza.

**Nomenclatura**

El idioma por defecto a la hora de dar sentido funcional al nombre de clases, variables, constantes, etc. será una mezcla entre la nomenclatura tradicional en inglés y la nomenclatura funcional adoptada.

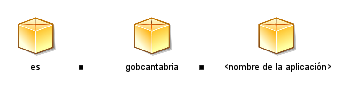
Resumiendo, aquella codificación que por estandarización y/o aceptación se pueda escribir en inglés se mantendrá así por convenio, casos como *insert*, *update*, *delete*, *create*, *retrieve*, *list*, *set*, *get*, *newInstance*, *Delegate*.

Para la parte funcional se utilizara castellano, por lo tanto la nomenclatura de los métodos será: *getListEmpresa* en sustitución de *getListCompany* o *insertBanco* en lugar de *insertarBanco*.

**Paquetes**

Por defecto todos los paquetes se escribirán en minúsculas y sin utilizar caracteres especiales. El paquete base queda definido como es.gobcantabria, en este paquete no se definirá ninguna clase.

Se tendrá, así mismo, otro nivel extra dentro del paquete definido como el nombre del proyecto o del módulo (Ej. es.gobcantabria.xxxx ).



Si existiera una parte común a varios de estos módulos, el nombre de los paquetes comenzarán por: *es.gobcantabria.common*

**Nombres de Interfaces**

Los nombres de interfaces utilizarán el sufijo **Interface** y estarán compuestos por palabras con la primera letra en mayúscula (CamelCase). Se debe evitar el uso de abreviaciones que dificulten la comprensión del código.

Ejemplo: *ConexionInterface*, *ComponenteTablaInterface*

**Nombres de clases**

Los nombres de clases deben ser mezclas de mayúsculas y minúsculas, con la primera letra de cada palabra interna en mayúsculas (CamelCase).

Debemos intentar mantener los nombres de clases simples y descriptivos.

Debemos usar palabras completas y evitar acrónimos y abreviaturas (se permiten DAO, DTO, URL, HTML, etc.).

Si la clase cumpliese algún patrón determinado o tuviese una funcionalidad específica es recomendable definirlo en el nombre.