gmail.com



CALIDAD EN EL DESARROLLO DEL SOFTWARE

**Estudiante(s):** Grecia Ignacia Cruz Góngora

Universidad Tecnológica Riviera Maya

T.S.U. en Tecnologías de la Información y Comunicación

**PLANEACION**

**OBJETIVO**

Crear una aplicación que permita que los usuarios registren clientes, añadiendo sus datos básicos, los cuales podrán tener una categoría, según la empresa donde se colabore.

**ALCANCE**

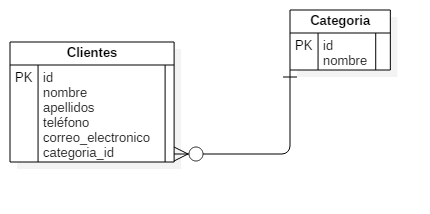
Desarrollar una aplicación donde el cliente pueda gestionar los datos del mismo.

**REQUERIMIENTOS**

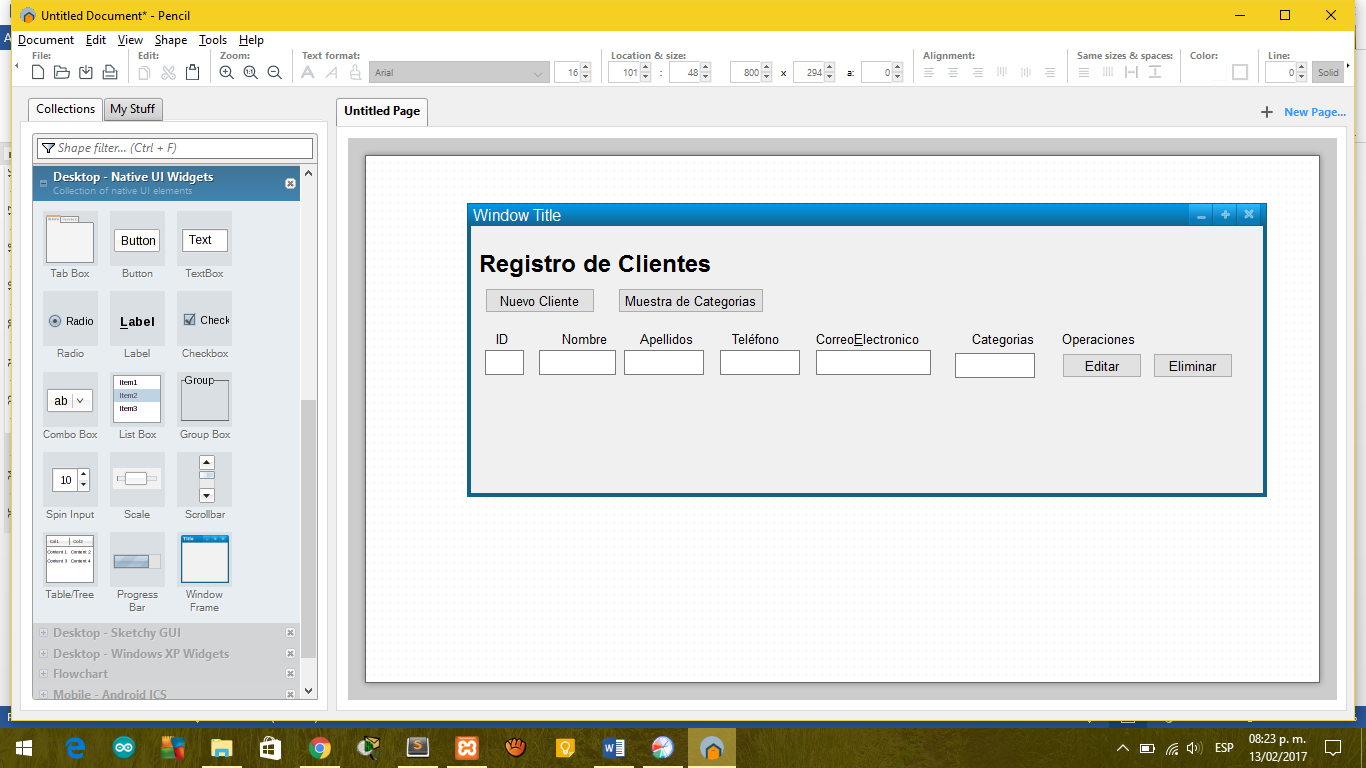
* Las categorías que estén relacionadas con un cliente no se podrán eliminar.
* Los nombres y apellidos serán obligatorios.
* Correos Electrónicos válidos.
* La aplicación será web
* El nombre de la categoría no puede quedar vacío.
* Los correos electrónicos no se pueden duplicar

**DISEÑO**

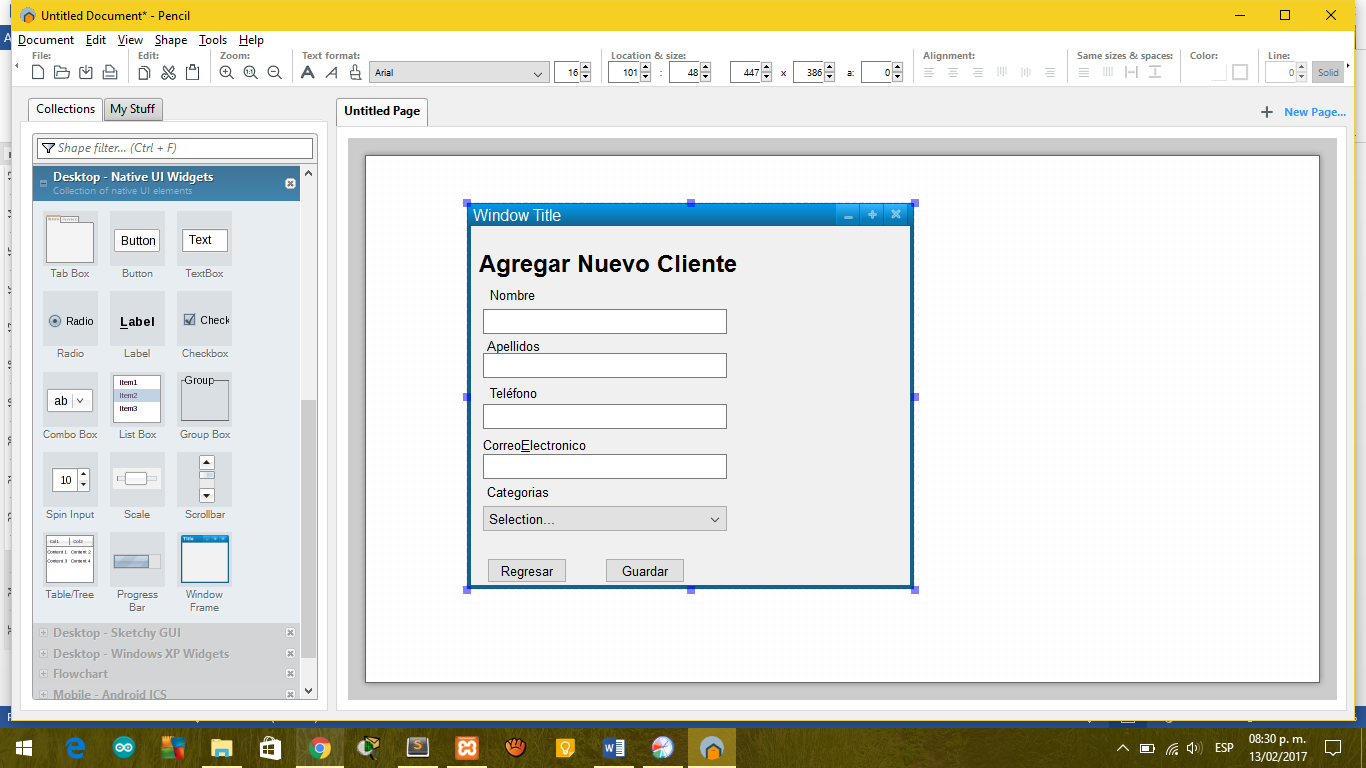
La base de datos será la siguiente la cual almacenará la información más relevante del cliente, al cual se asignara una categoría según los datos obtenidos.

****

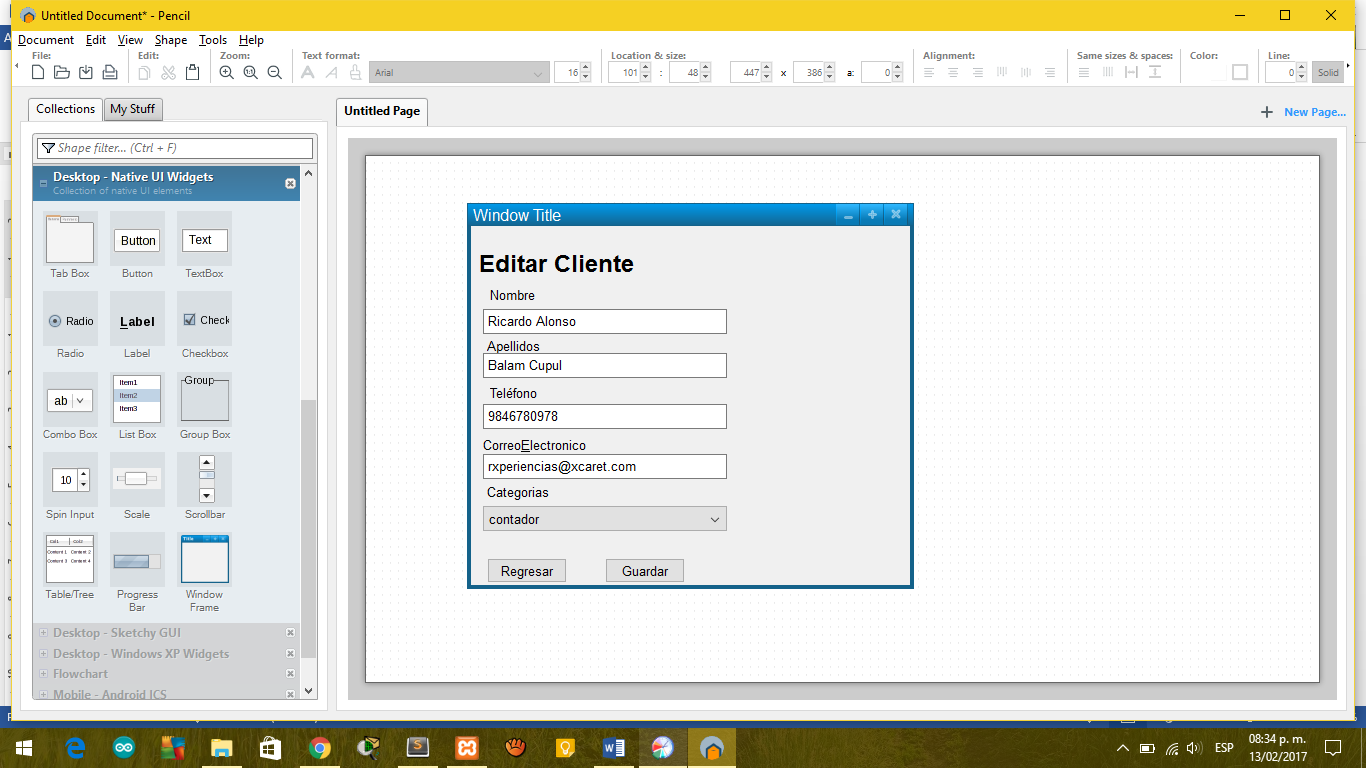
Interfaz de la página web (INICIO)



Botón nuevo cliente

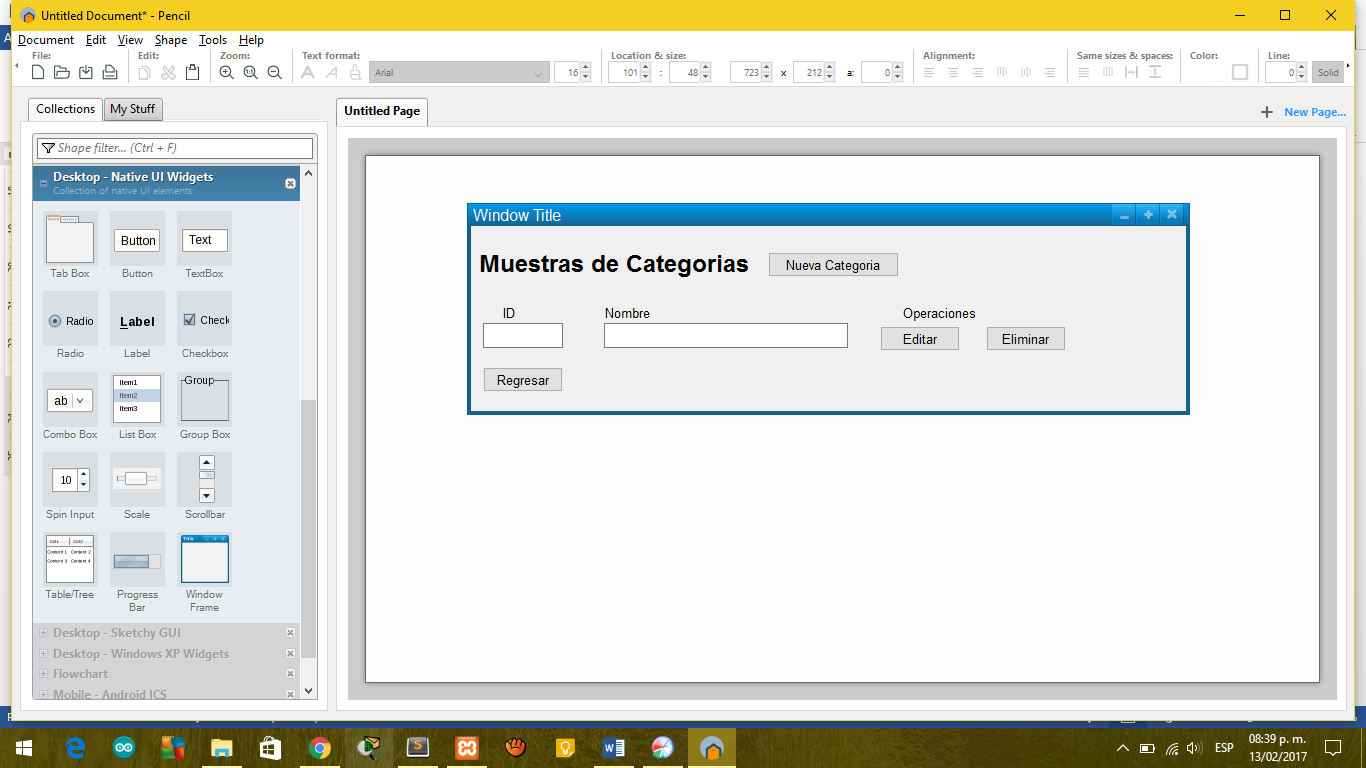


Botón editar cliente



El botón eliminar dará click y se eliminara

Boton muestra de categorias



Definición de línea de código

* **Convencion de Nombres**

Se empleara un escrito de minusculas todas,en el caso de tener dos o mas palabras a nombrar quedaria de la siguiente manera: primera palabra en mayuscula y la segunda palabra que lo acompre la primera letra sera mayuscula y el resto minuscula.

Ejemplo:

**1: publico**

**2:publicoSocial**

* **Nombre de Archivos**

Las carpetas seran nombradas en minusculas debido al rapido manejo que tiene el programador a la hora de la localizacion de cada archivo.

* **Nombre de Clases**

Se empleara la primera letra en minuscula, esto aplicara a todas las clases en el codigo. En caso de que la clase sea de dos palabras se realizara lo mismo para ambas palabras iniciando con la letra mayuscula y lo demas minusculas.

* Ejemplo de Clases
* **class publico{**

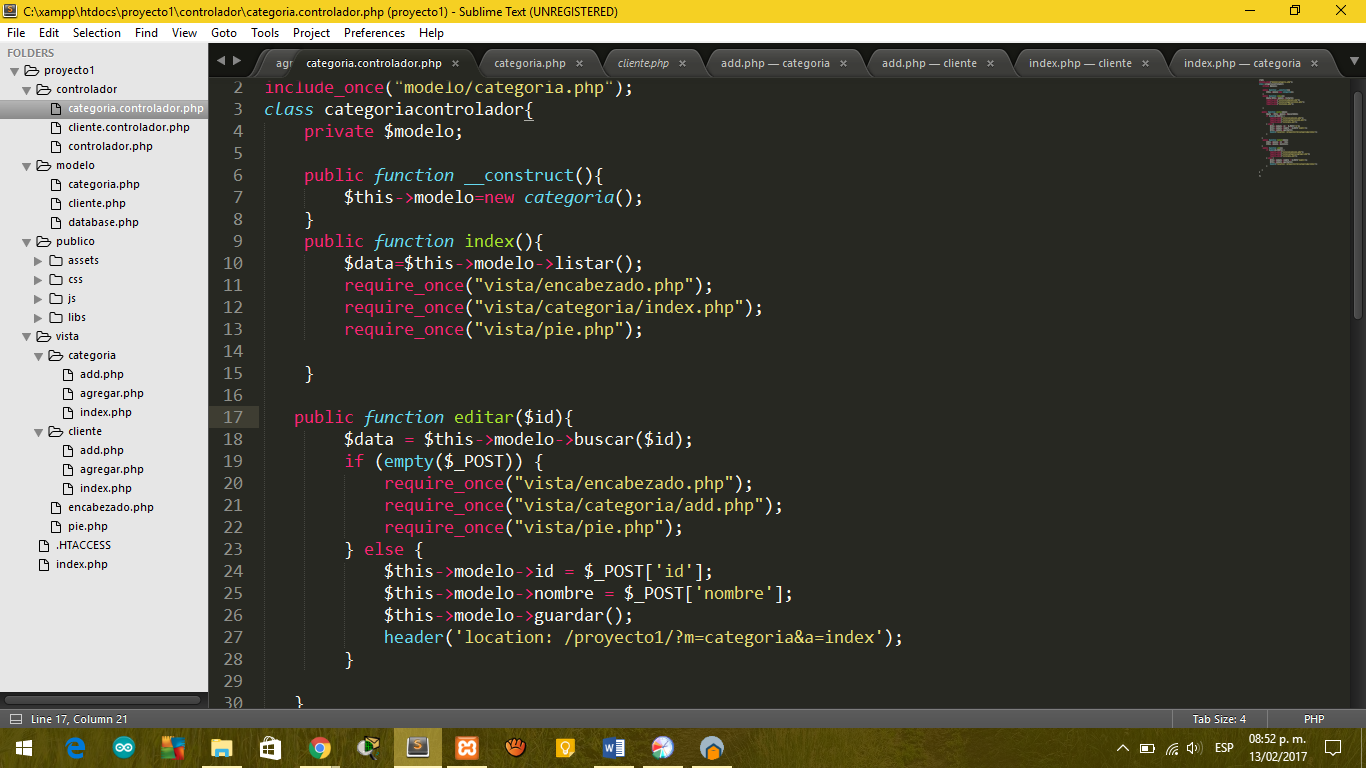
**}**

* **class PublicoSocial{**

**}**

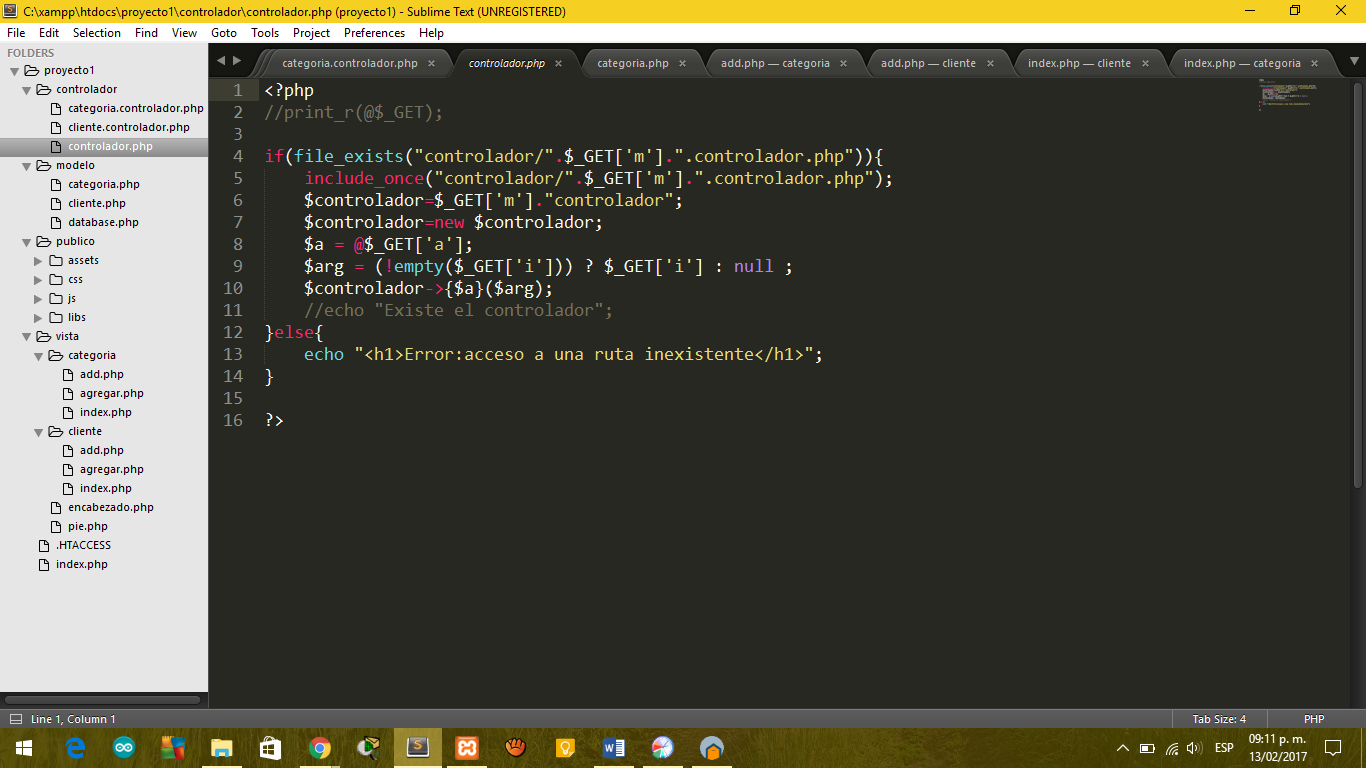
* **Nombre de Funciones/métodos**

Deben ser escritos utilizando la técnica camelCase ( lowerCamelCase, la primera palabra es minúscula y si hay una segunda debera ser la primeraletra Mayuscula y las demas minusculas.

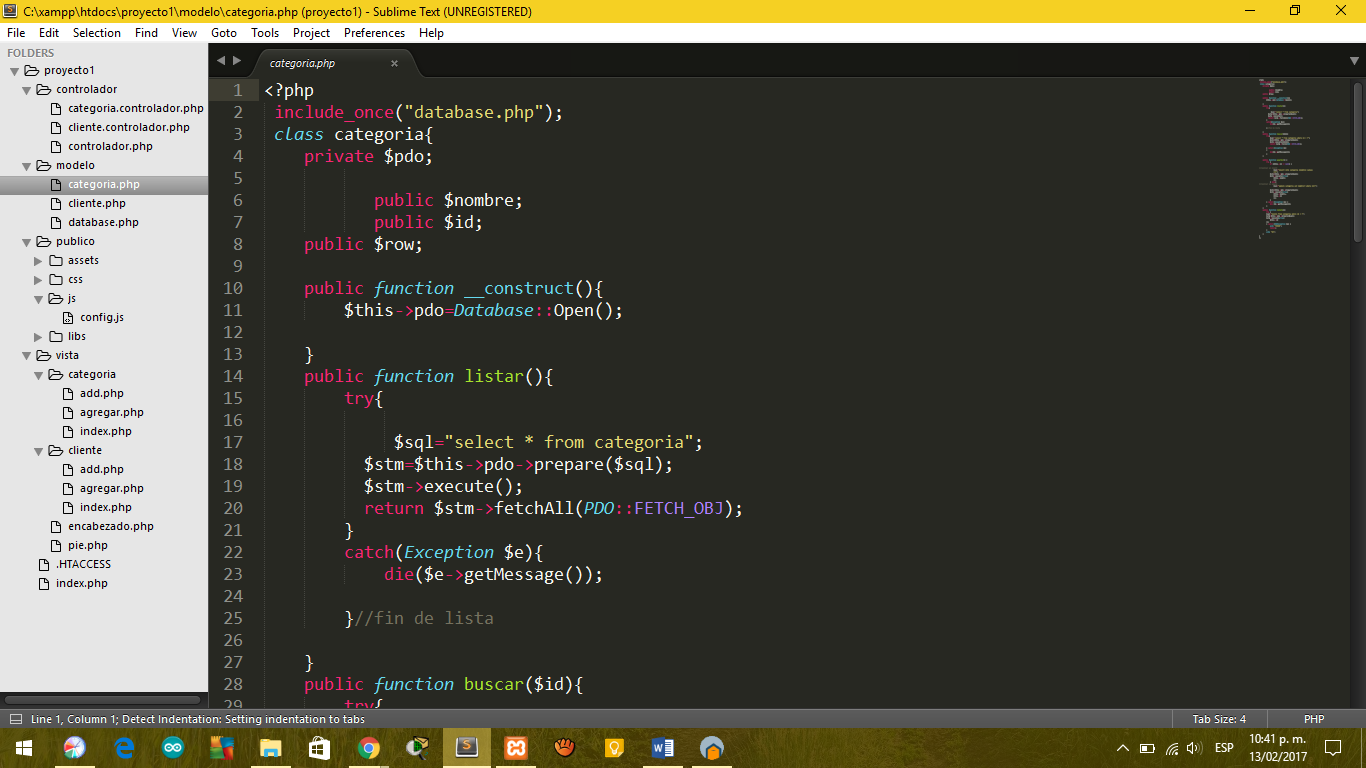
* Ejemplo de Funciones
* **Nombre de Variables/propiedades**

Estaran escritas en minusculas aun siendo la primera letra en dado caso de aver una segunda palabra, su primera letra sera en mayuscula y las demas en minusculas haciendo un guion bajo entre las dos palabras.

* Ejemplo de Variables



Nose debe utilizar la palabra reservada var para declarar una propiedad, debes utilizar public, private, static o protected.



* **Nombre de Constantes**

Se mostraran en minusculas las palabras no habra cambios en ella,asi es mas facil identificarlas.

* Ejemplo de Constante

**Const version = ‘1.0’;**

* **Nombre de las Tablas de la Base de Datos**

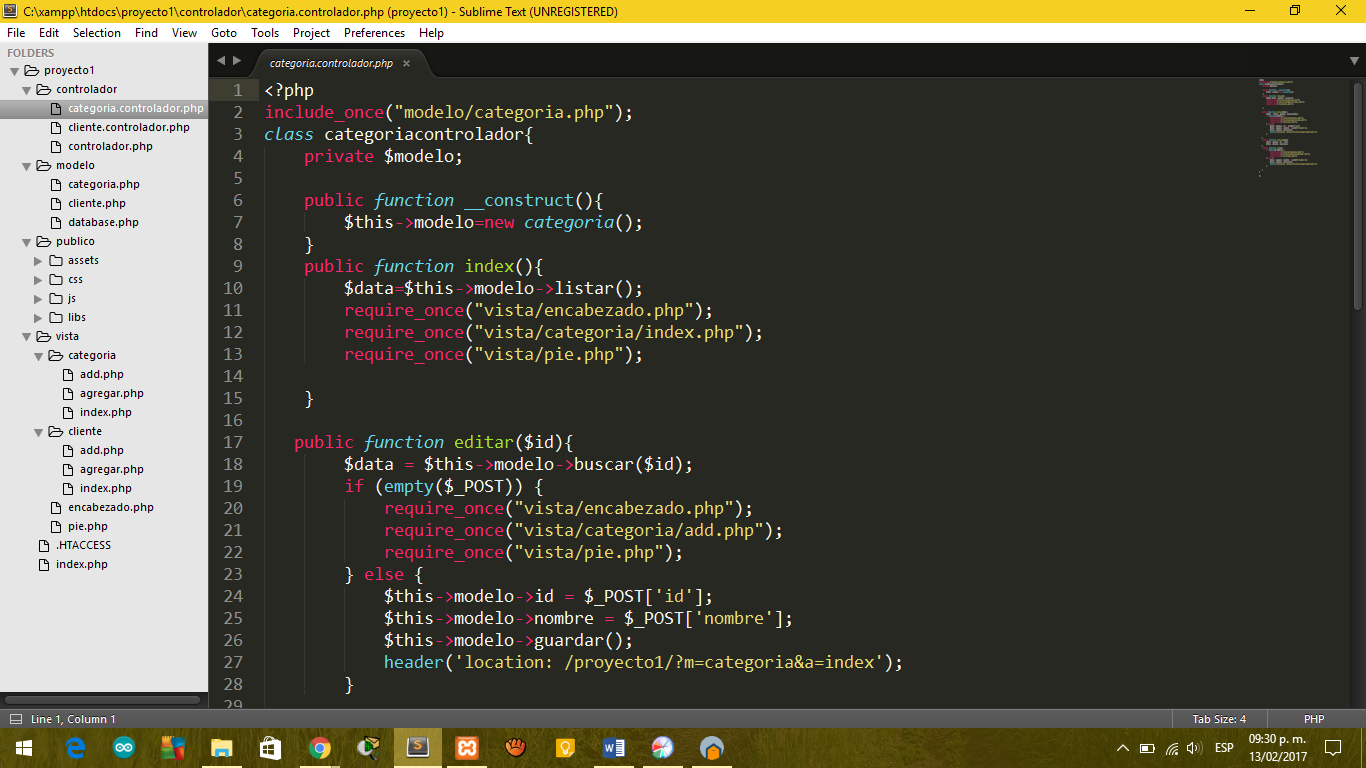
Las tablas en la base de datos seran escritas en minusculas.

* **Estructura de Control**

Debe haber un espacio después de cada estructura de control (if, foreach, try).

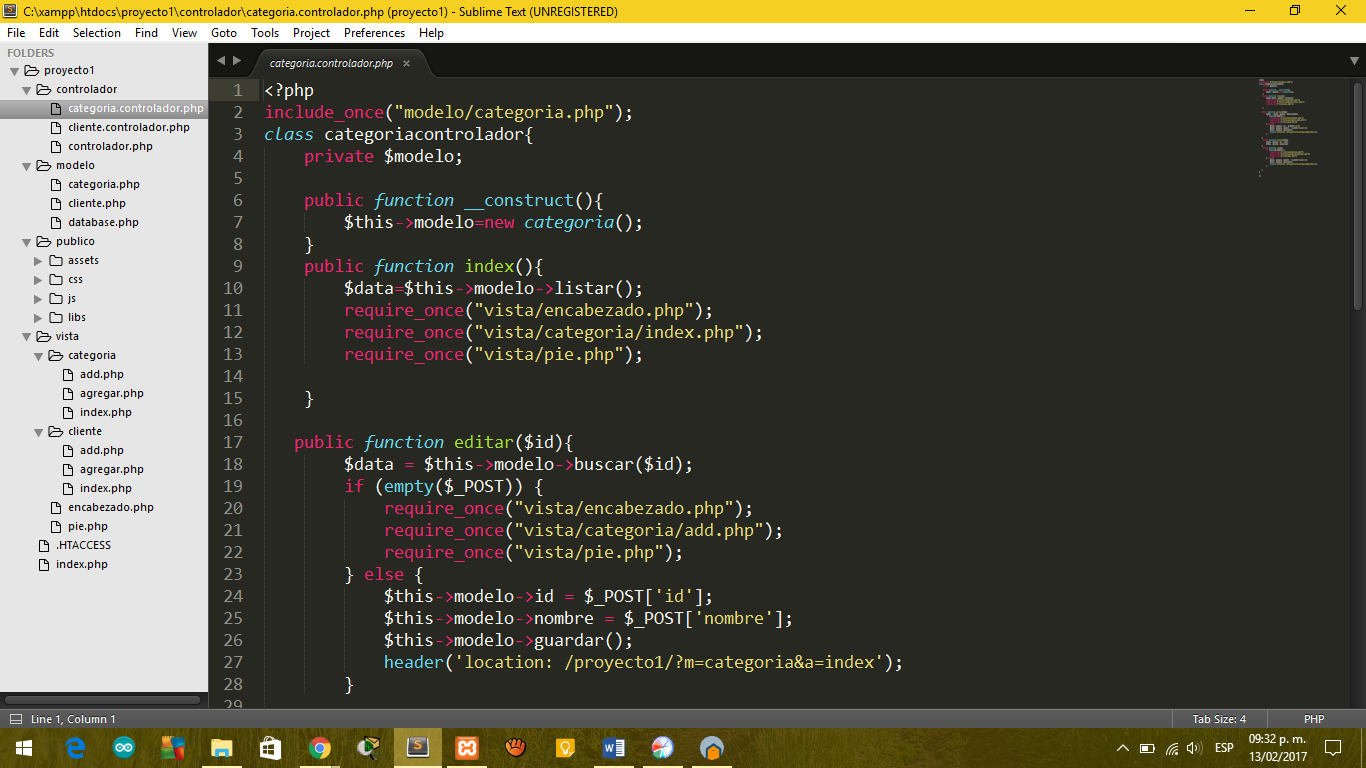
* If

La sentencia IF sera escrito en minusculas (if).



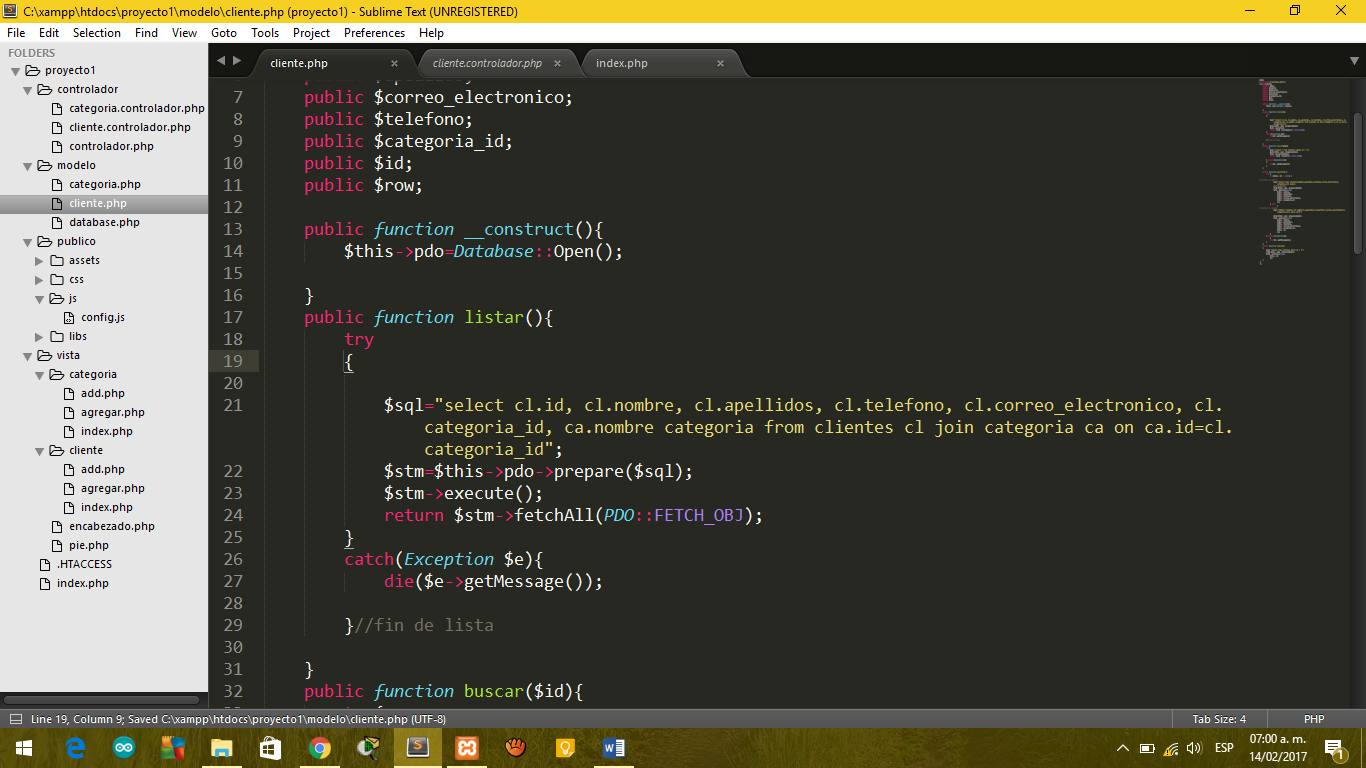
* Else

La sentencia ELSE sera escrito en minusculas (else).



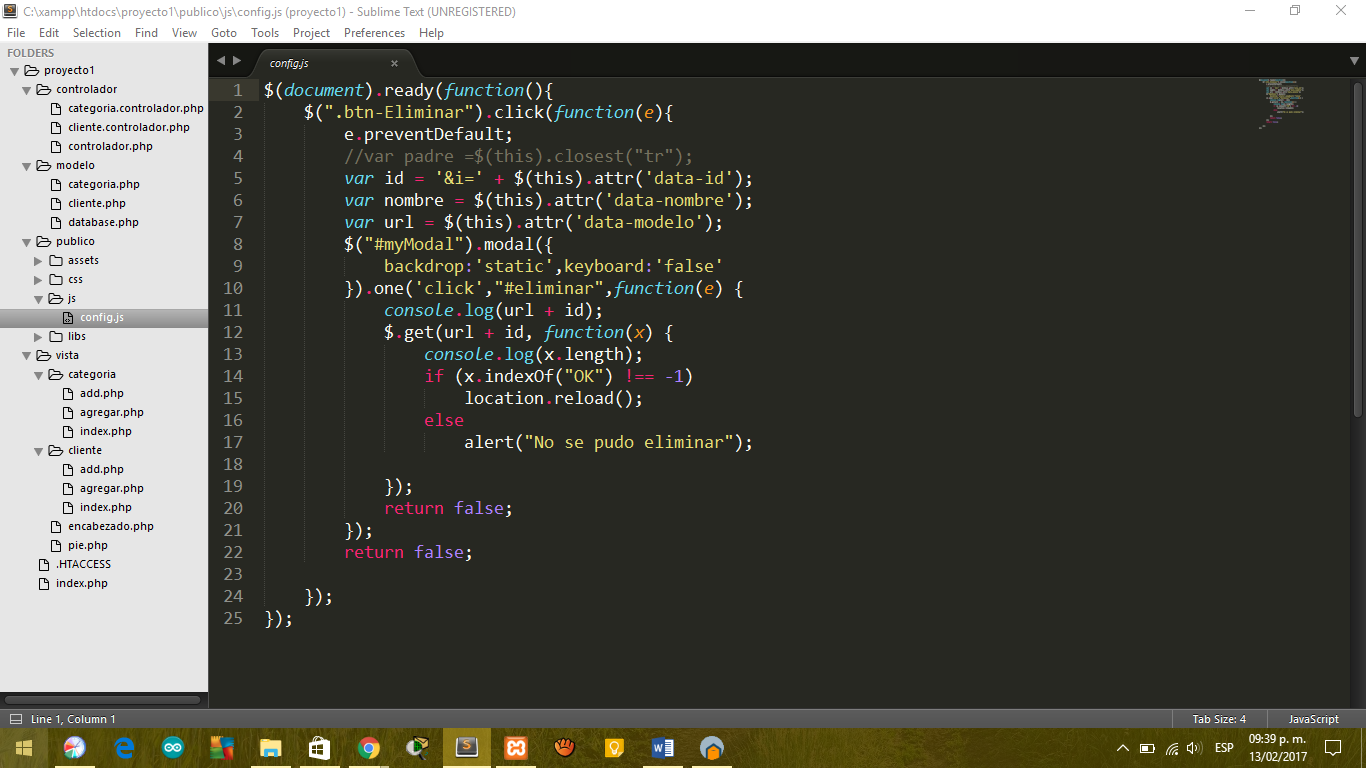
* Try

La sentencia TRY sera escrito en minusculas (try) y tendra un espacio despues de poner las llaves.

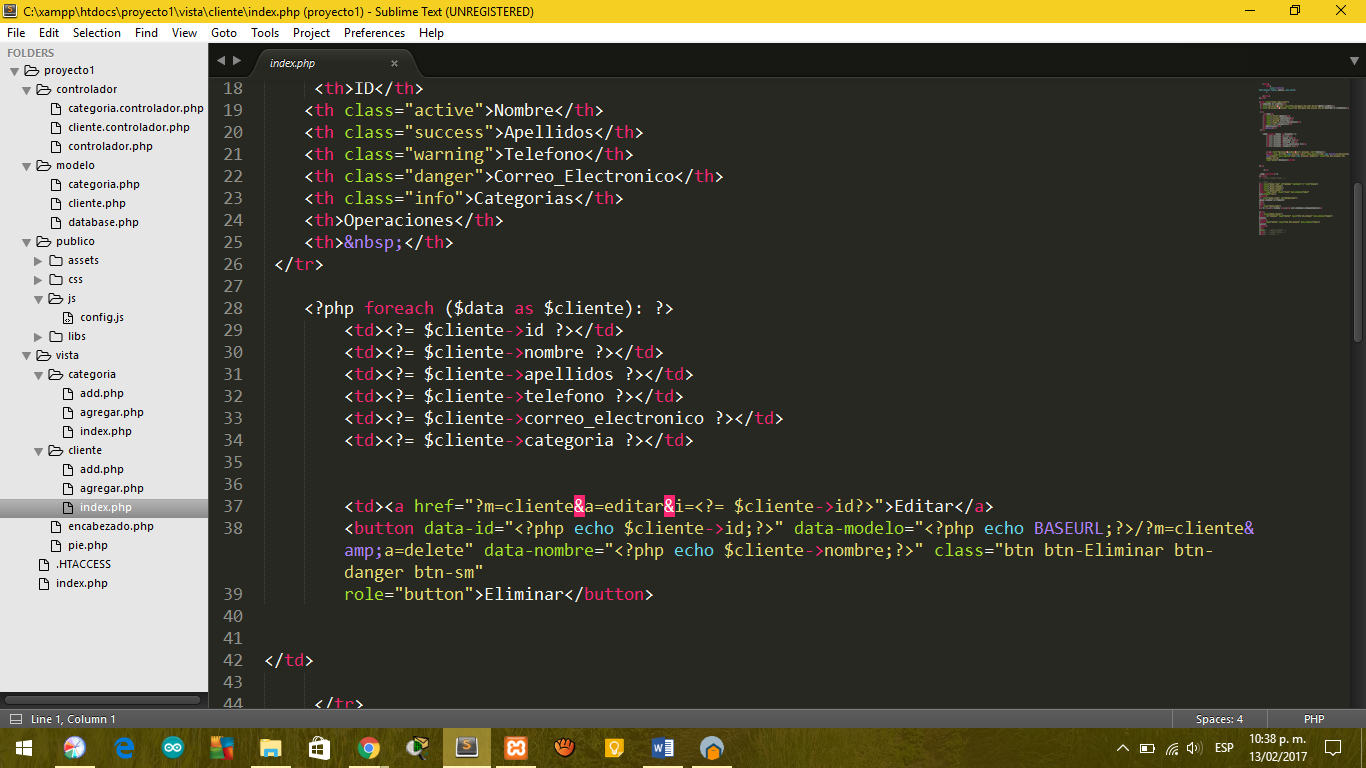


* Operador ternario

Estos tipos de operadores evaluan una condicion y retornan un valor dependiendo si la condicion es verdadera o falsa.



* **Estructura loops/herativas**
* Foreach

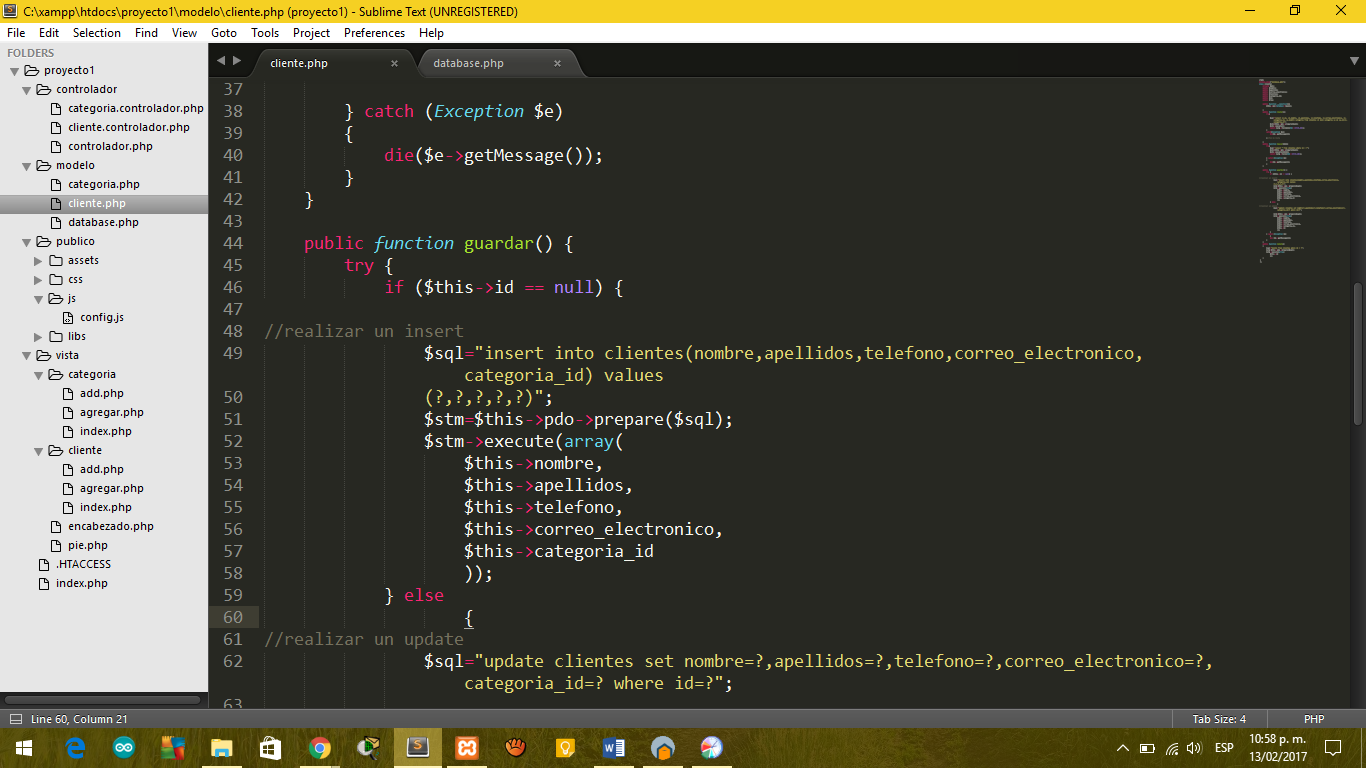


* **Comentarios**

Todas las funciones deben tener un comentario, antes de su declaración, explicando que hacen.

Se empleran // para los comentarios que no exceden a un parrafo, y para los que exceden se empleran los /\*\*/ para describir mas a fondo el método.

* Ejemplos



**COMPILACIÓN**

En el caso de esta aplicación web no es necesario la compilación.

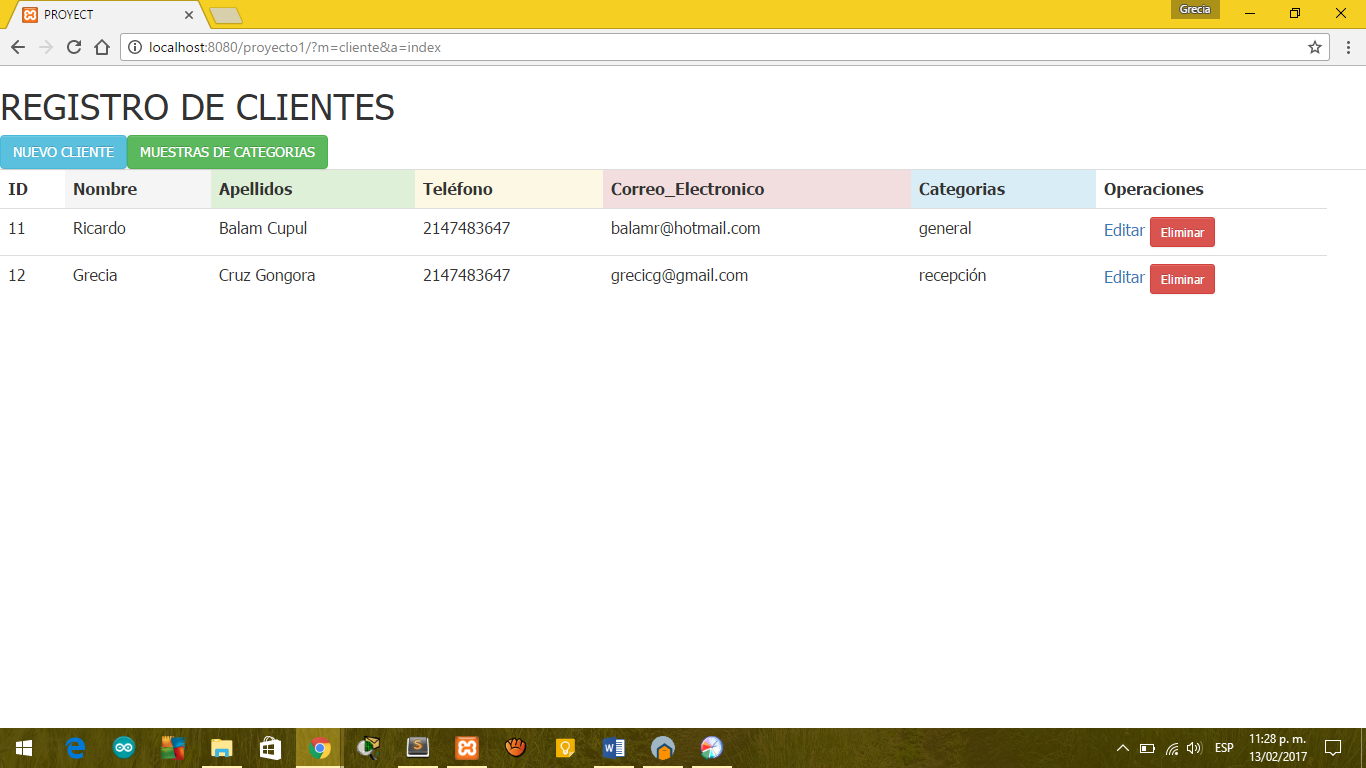
**PRUEBAS**

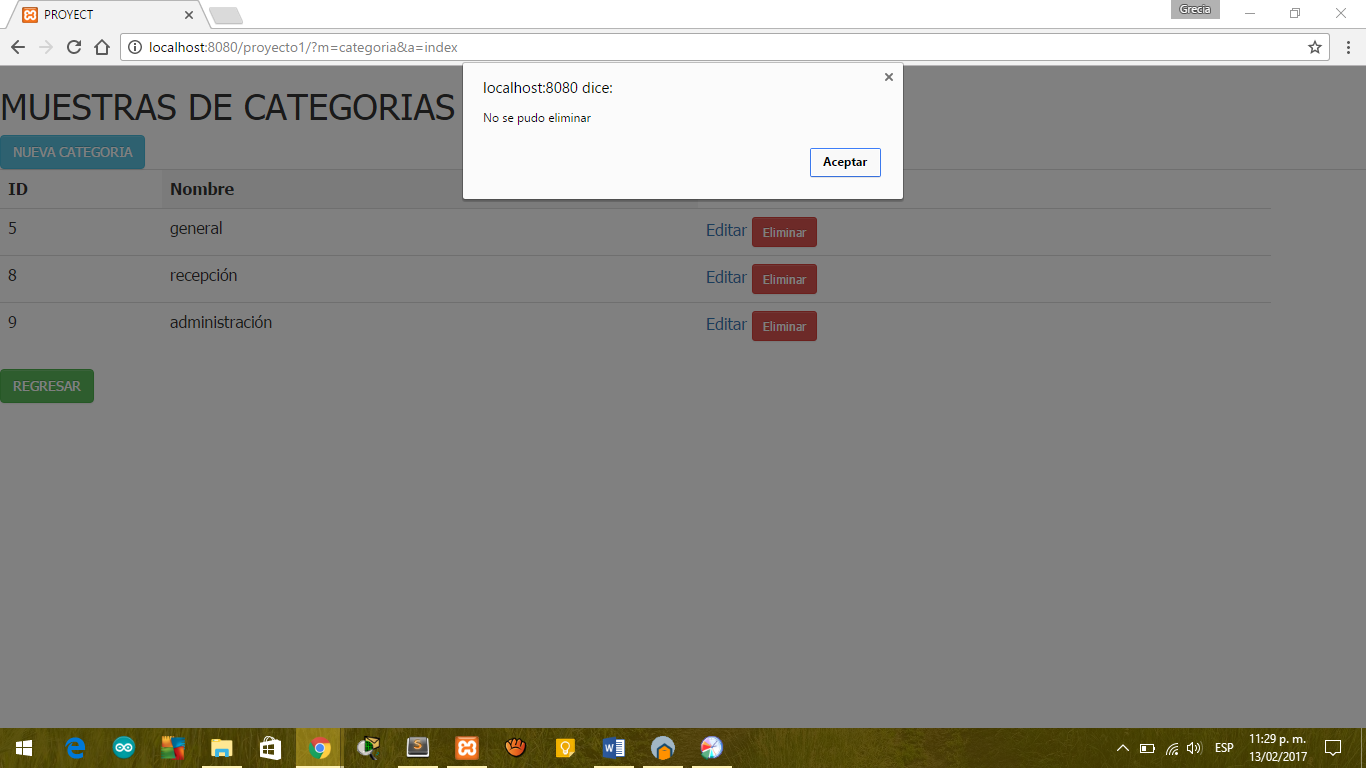
1. Al ejecutar la aplicación se encontro como primer error que las categorias no se enlazaban automaticamente con la tabla de clientes, impidiendo asi saber cuales categorias se habian añadido.

2. Se corrigio el primer error pero se encontro que al eliminar una categoria se eliminaba el cliente y esto ocasionaba que todos los clientes con esa categoria ya no existian.

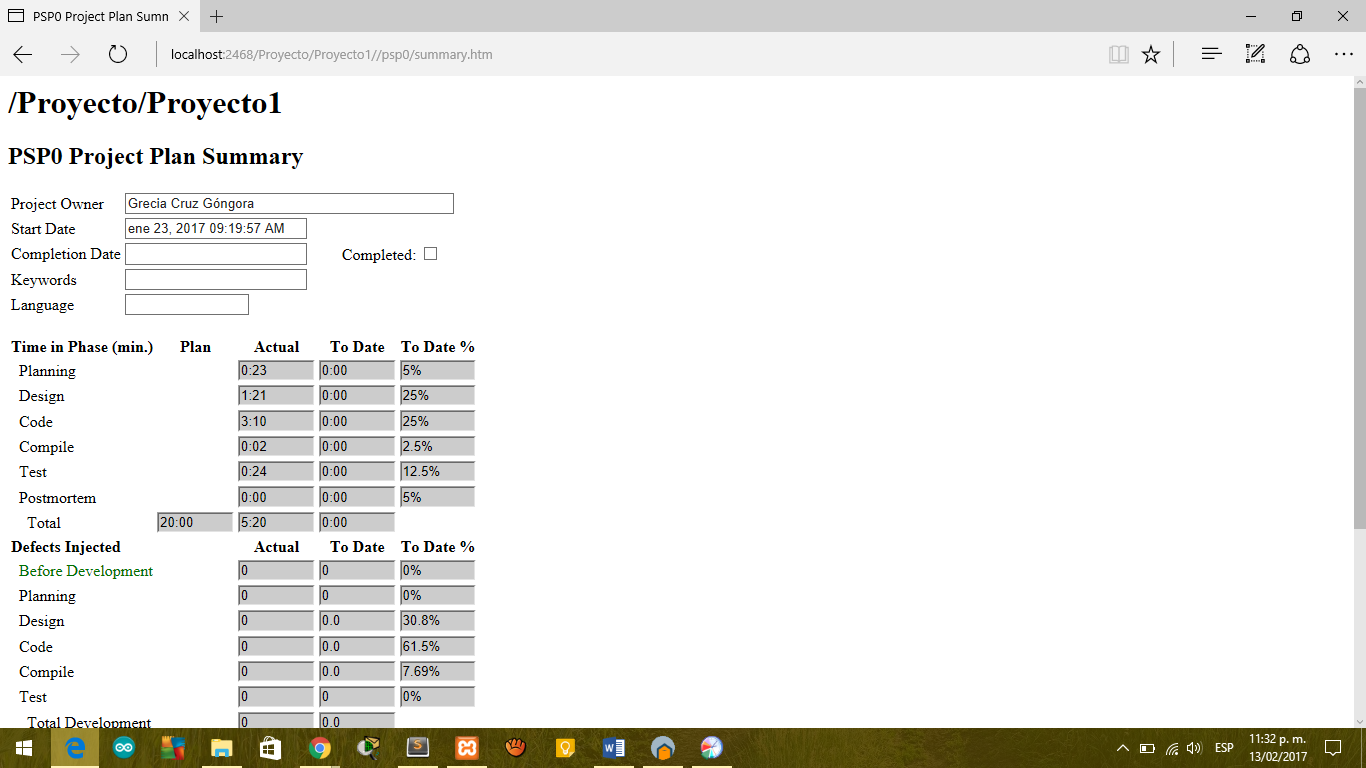
3. Al probarlo algunas personas, se encontro que los ID no eran automaticos entonces habia un error en la base de datos.

4. Y por ultimo asi quedo con los errores ya corregidos.





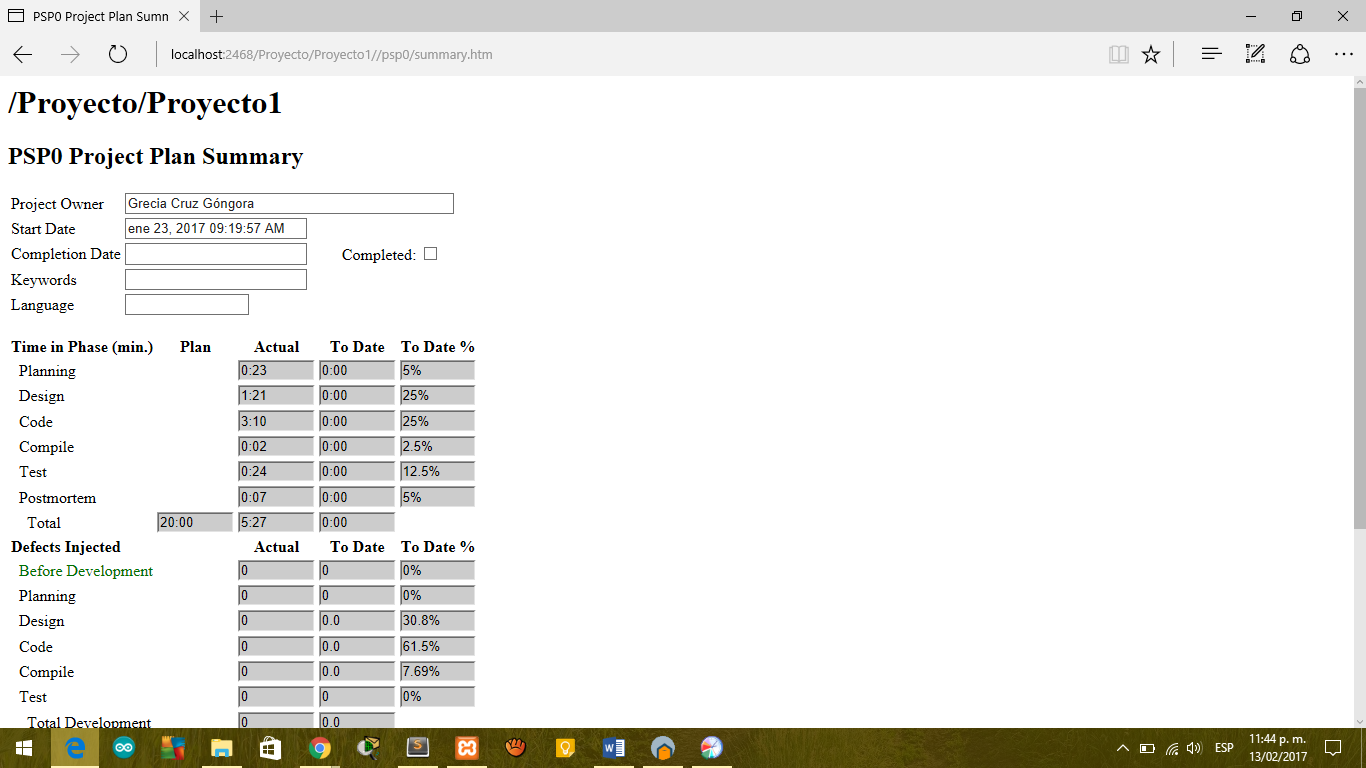
Se finalizó con esto la base del desarrollo y se obtuvo lo siguiente.



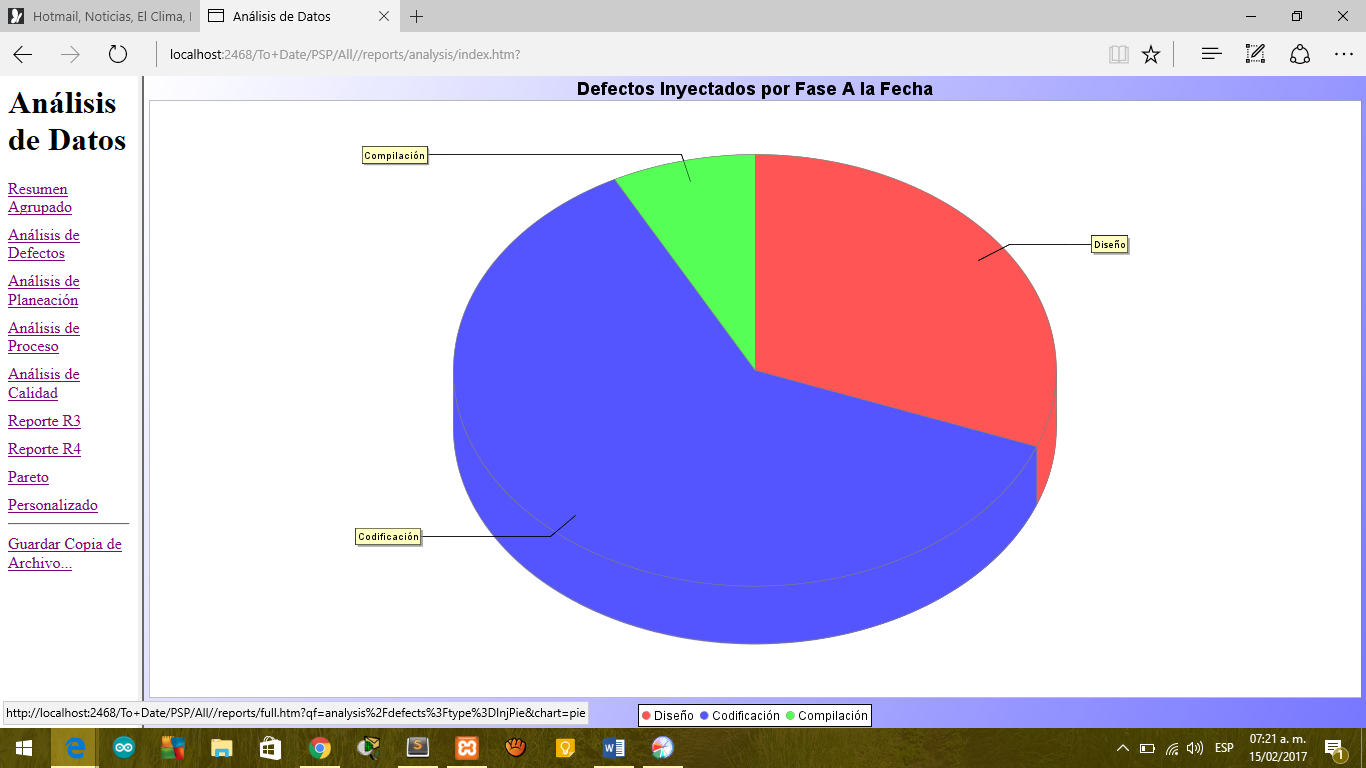
**POSTMORTEN**

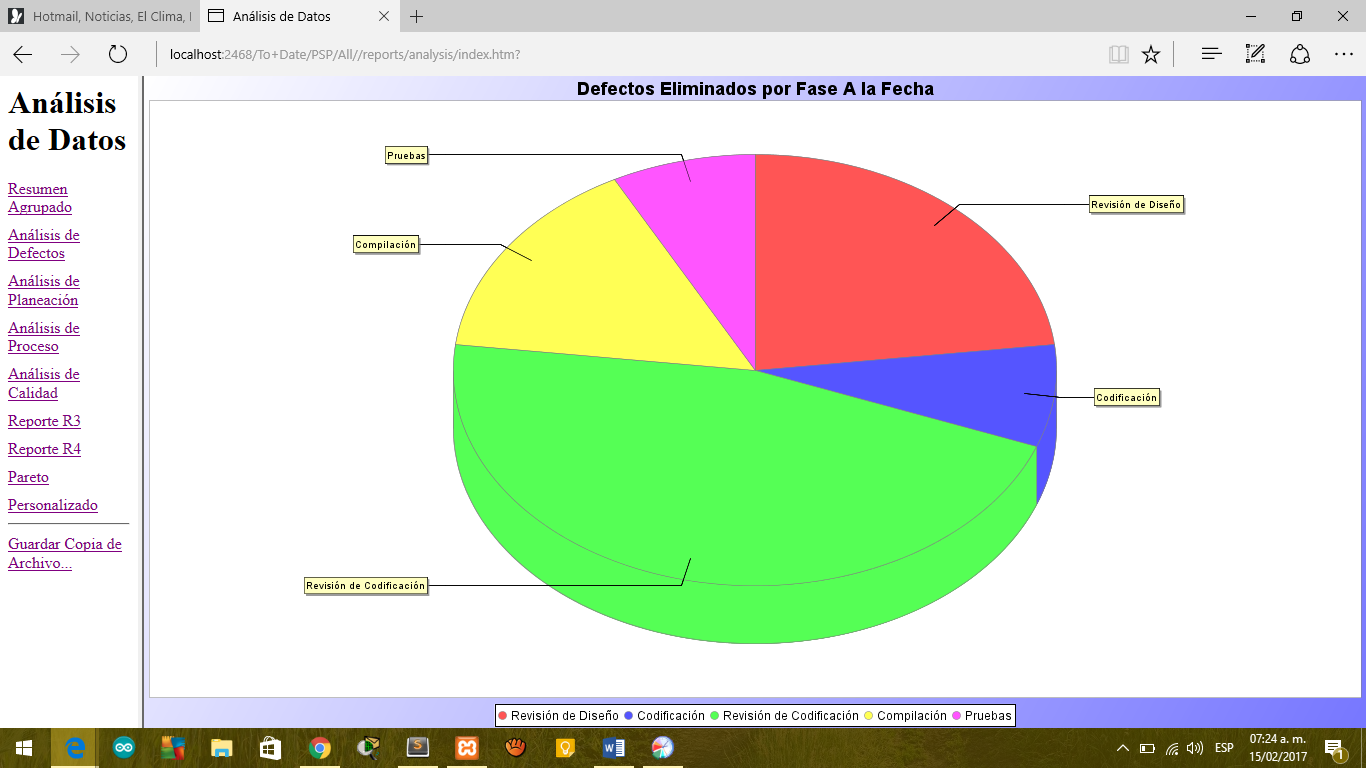
En esta fase lo importante es verificar la consistencia la cual para ello se realizó la verificacion de los datos fueran los correctos y si se tiene un uso adecuado.

Asi mismo el funcionamiento de los botones como dice el nombre que sea la peticion que realice.



**REPORTES DEL SOFTWARE PROCESS DASHBOARD**





**Bitácora de tiempos y defectos de Process Dashboard.**

