Nomenclatura

## Nombres de los paquetes

App\_Data  
Contendrá los archivos de bases de datos .mdf

Paquetes

### Acceso a datos:

Contendra el archivo DBML (data context) que hace el mapeado de la base de datos en objetos. Las clases que contendrán todas las clases de acceso a datos se llamaran objetoRepository, y de ser necesario se paquetes para unir clases con objetos comunes.

Ejemplo:   
El método para hacer la validación de usuario durante una identificación seria accesado por la siguiente ruta

*Modelos.usuarioRepository.Validar(user);*

### Presentación:

Serán almacenadas en la raíz del proyecto y en caso de haber paginas dependientes serán guardadas en una carpeta con el mismo nombre, por ejemplo http://localhost:4143/Clientes/agregar.aspx, dichas paginas siempre serán llamadas en minúsculas y en caso de formarse de 2 o más palabras separadas con guion bajo.

### Controles:

Contendrán las clases que controlaran el flujo entre la capa de presentación y la capa de acceso a datos dichas clases serán llamadas objetoController y de ser necesario se paquetes para unir clases con objetos comunes.

Ejemplo:  
El método a ejecutar durante la validación de un usuario seria

*UsuarioController.Login(usuario);*

No utilizamos el nombre del paquete por que estos métodos son accesados desde archivos .aspx

## Codificación

### Clases

* Los nombres de las clases siempre serán con respecto a los objetos.
* Comenzaran con letra mayúscula.
* Todas restricciones del lenguaje (como números y caracteres especiales).

### Variables y métodos

Se utilizara la nomenclatura CamelCase que facilita la lectura del código.

#### Variables:

* Primera letra minúscula
* Si está formada por 2 o más palabras se utilizaran mayúsculas (ejemplo: claveDeCliente).
* Se procurara utilizar lo menos números posibles siempre tratando de definir nombres claros para los desarrolladores (ejemplo: cvdp2 por querer simplificar claveDelProducto2).
* Las limitaciones del lenguaje (números y caracteres especiales).

#### Constantes

* Siempre en letras mayúsculas (ejemplo: IVA en caso de ser algún acrónimo comentar el significado //Impuesto del valor agregado).
* Las limitaciones del lenguaje (números y caracteres especiales).

#### Propiedades

* Empezaran con guion bajo y utilizando CamelCase (ejemplo: \_edad, \_numeroTelefonico).
* Siempre serán privadas.
* Las limitaciones del lenguaje (números y caracteres especiales).

#### Métodos

* El inicio de cada palabra siempre será en mayúsculas (ejemplo: Login(), CrearUsuario(user), BuscarUsuario(Id)).
* Las limitaciones del lenguaje (números y caracteres especiales).
* Los parámetros de entrada serán nombrados en lower camelcase (ejemplo BuscarUsuario(int id)), para identificarlo de las propiedades y accesores.