Este documento representa la declaración del estándar de codificación que se implementara al momento de codificar el proyecto ADeux.

**Convención de nombres para el código**

* **Variables**
* Los nombres de las variables se escribirán en minúsculas.
* Los nombres de las viables que se compongan de dos o más palabras se escribirán en formato Ajunto.
* Para los nombres de las variables no se usaran caracteres especiales.
* Los nombres de las variables solo se escribirán con letras y numeros del a-z y del 0-9.
* **Constantes**
* Los nombres de las constantes se escribirán en mayúsculas y/o minuscula.
* Los nombres de las constanes solo se escribirán con letras y umeros del A-Z y del 0-9.
* Para los nombres de las constantes no se usaran caracteres especiales.

**Convención de nombres para las tablas de la base de datos**

* Se utilizara caracteres alfanuméricos.
* El limite los nombres a menos de 64 caracteres (limitación de MySQL).
* Se utilizara el guion bajo (\_) para separar palabras o adjunto simpre y cuando sea en letra mayuscula la primera letra de cada palabra.
* Se utilizara palabras en minúsculas.
* Se utilizara nombres en el lenguaje inglés y español.
* Los nombres de las tablas deben ir en plural y los nombres de las columnas en singular.
* Se utilizara las letras id en las columnas de clave principal y externa.
* En una tabla, se utilizara colocara en primer lugar la clave principal, seguida de las claves externas.
* Los nombres de los campos deben ser descriptivos de su contenido.
* Los nombres de los campos deben ser unívocos entre las tablas, excepción hecha en las claves.

**Inclusión de archivos**

Para incluir archivos se usara la siguiente función.

require\_once

**Comentarios**

Los comentarios se escribirán de la siguiente forma:

**Comentarios de una sola línea**

//Este es un comentario de una línea

**Comentarios de dos líneas o más**

/\*\*

\* Este es un comentario de más de una línea

\*/

**Estructuras de control**

Las estructuras de control se escribirán de la siguiente manera

**Estructura if**

<?php

if ((condition1) || (condition2)) {

action1;

} elseif ((condition3) && (condition4)) {

action2;

} else {

defaultaction;

}

?>

**Estructura switch**

switch (condition) {

case 1:

action1;

break;

case 2:

action2;

break;

default:

defaultaction;

break;

}

**Dividir sentencias if largas**

if (($condition1 || $condition2) && $condition3 && $condition4) {

//code here

}

**Operador ternario**

<?php

$a = $condition1 && $condition2

? $foo : $bar;

$b = $condition3 && $condition4

? $foo\_man\_this\_is\_too\_long\_what\_should\_i\_do

: $bar;

?>