

# API Documentation

## Documentation

### PACKAGES

 [Default](#)

### CHARTS

 [Class inheritance diagram](#)

### REPORTS

 [Errors](#) **0**

 [Markers](#)

 [Deprecated elements](#) **0**

Template is built using [Twitter Bootstrap 2](#) and icons provided by [Glyphicons](#).

Documentation is powered by [phpDocumentor 2.8.2](#) and

generated on 2015-03-25T13:12:57-05:00.



## PACKAGES

 *Default*

## CLASSES

*User\_model*

*User*

*User\_standard\_model*

*User\_standard*

 \ Default

## Classes, interfaces, and traits

### User

Esta clase se encarga de generar formularios automaticos a partir de

### User\_model

Clase que contiene las funcionalidades de una usuario no registrado

### User\_standard

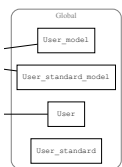
Clase que maneja el estandar dado por el Usuario

### User\_standard\_model

Esta clase contiene los metodos de insercion actualizacion y borrado de metadatos pertenecientes al estandar proporcionado por el usuario Class User\_standard\_model

Template is built using [Twitter Bootstrap 2](#) and icons provided by [Glyphicons](#).

Documentation is powered by [phpDocumentor 2.8.2](#) and  
generated on 2015-03-25T13:12:57-05:00.



Template is built using [Twitter Bootstrap 2](#) and icons provided by [Glyphicons](#).

Documentation is powered by [phpDocumentor 2.8.2](#) and

generated on 2015-03-25T13:12:57-05:00.



PUBLIC

PROTECTED

PRIVATE

INHERITED

**M** METHODS

- *Constructor de la clase que carga las librerias encrypt y user\_standard, ademas de crear una nueva estancia de la clase user\_standard*

`__construct()`

- *Funcion se encarga de borrar los metadatos de cada categoria del estandar brindado por el usuario*

`delete_metadato()`

- *Es el metodo por defecto donde se carga las paginas*

`index()`

- *Funcion se encarga de guardar los datos de cada categoria del estandar brindado por el usuario*

`save_category()`

- *Esta función se encarga de subir archivos mediante el metodo ajax solo para un archivo*

`uploadfile()`

- *Esta clase se encarga de mostrar la vista donde se carga el formulario dinamico apartir del estandar proporcionado por el usuario*

`wizard_form()`**P** PROPERTIES**C** \ \ User

Esta clase se encarga de generar formularios automaticos a partir de

Access	public
Autor	Danny Alexander Trujillo Pulgarin
Category	Usuarios_No_Registrados
Package	Controladores
Subpackage	Controladores/Usuario Class User
Package	Default

**M** Methods

Constructor de la clase que carga las librerias encrypt y user\_standard, ademas de crear una nueva estancia de la clase user\_standard

`__construct()`

Funcion se encarga de borrar los metadatos de cada categoria del estandar  
brindado por el usuario

```
delete_metadato()
```

Es el metodo por defecto donde se carga las paginas

```
index()
```

Funcion se encarga de guardar los datos de cada categoria del estandar brindado  
por el usuario

```
save_category()
```

Esta función se encarga de subir archivos mediante el metodo ajax solo para un  
archivo

```
uploadfile()
```

Esta clase se encarga de mostrar la vista donde se carga el formulario dinamico  
apartir del estandar proporcionado por el usuario

```
wizard_form()
```

## Properties

**PUBLIC**

PROTECTED

PRIVATE

INHERITED

 \ \ User\_model

Clase que contiene las funcionalidades de una usuario no registrado

Class User\_model

**Package**

Default

Template is built using [Twitter Bootstrap 2](#) and icons provided by [Glyphicons](#).

Documentation is powered by [phpDocumentor 2.8.2](#) and  
generated on 2015-03-25T13:12:57-05:00.



PUBLIC

PROTECTED

PRIVATE

INHERITED

**M) METHODS**

- *Esta funcion crea el objeto CI de la clase codeigniter e invoca el modelo user\_standard\_model*

```
__construct()
```

- *Borra filas de registros correspondientes al id del OA*

```
delete_in_table()
```

- *Esta clase consulta el ultimo id guardado de los OA's y entre el mismo sumado 1*

```
get_actual_id()
```

- *Retorna array con los hijos del estandar del usuario*

```
get_hijos()
```

- *Retorna los padres que tienen hijos y a su vez padres del estandar del usuario*

```
get_padres()
```

- *Esta función busca los padre que no tienen mas padres*

```
get_spadres()
```

- *Inserta en la tabla OAS registro en caso de que no exista el consecutivo del OA y en caso de que exista actualiza las columnas de esa fila*

```
insert_in_oas()
```

- *Esta funcion insertara o*

```
insert_in_table()
```

- *Esta funcion estructura el estandar proporcionado por el usuario identificando los padres que no tienen padres, padres con padres e hijo con padres*

```
structure_standard()
```

**P) PROPERTIES****C) \ \ User\_standard**

Clase que maneja el estandar dado por el Usuario

Autor	Danny Alenxander Trujillo
Package	Libreria
SubPackage	Estandar Usuario Class User_standard
Package	Default

**M) Methods**

Esta funcion crea el objeto CI de la clase codeigniter e invoca el modelo  
user\_standard\_model

```
__construct()
```

Borra filas de registros correspondientes al id del OA

```
delete_in_table($padre, $oa, $orden)
```

Esta clase consulta el ultimo id guardado de los OA's y entre el mismo sumado 1

```
get_actual_id() : integer
```

Retorna array con los hijos del estandar del usuario

```
get_hijos() : mixed
```

Retorna los padres que tienen hijos y a su vez padres del estandar del usuario

```
get_padres() : mixed
```

Esta función busca los padre que no tienen mas padres

```
get_spadres() : mixed
```

Inserta en la tabla OAS registro en caso de que no exista el consecutivo del OA y en caso de que exista actualiza las columnas de esa fila

```
insert_in_oas($padre, $hijo, $oa, $dato)
```

Esta funcion insertara o

```
insert_in_table($padre, $hijo, $oa, $dato, $orden)
```

Esta funcion estructura el estandar proporcionado por el usuario identificando los padres que no tienen padres, padres con padres e hijo con padres



```
structure_standard()
```

## Properties

---

Template is built using [Twitter Bootstrap 2](#) and icons provided by [Glyphicons](#).

Documentation is powered by [phpDocumentor 2.8.2](#) and  
generated on 2015-03-25T13:12:57-05:00.

≡

≡

PUBLIC

PROTECTED

PRIVATE

INHERITED

**M** METHODS

- *Función que borra la fila de una tabla identificando su consecutivo*

```
delete_metadato_oa_table()
```

- *Se revisa que exista al menos una fila en la tabla*

```
exists_insert_oa()
```

- *Esta función busca si existe registrado al menos el oa correspondiente a una tabla*

```
exists_oa_table()
```

- *Retorna el consecutivo inmediatamente siguiente al ultimo id del oa que se tiene*

```
get_last_row()
```

- *Esta función devuelve el metadato categoria resultado de la busqueda en el estandar del usuario*

```
get_metadato_father()
```

- *Funcion que obtiene el parent id del padre*

```
get_parentid()
```

- *Función que busca y retorna la estructura el estandar de metadatos proporcionado por el usuario*

```
get_user_standard()
```

- *En caso de no existir fila con el consecutivo del oa se insertara los datos*

```
insert_metadato_oa_suppadre()
```

- *Esta función inserta datos en una tabla determinada, identificandola con los parametros de superpadre, padre e hijo y este se asocia al id del objeto de aprendizaje*

```
insert_metadato_oa_table()
```

- *En caso de que exista algun dato se actualizara la fila*

```
update_metadato_oa_suppadre()
```

- *Esta función actualizara datos en una tabla determinada, identificandola con los parametros de superpadre, padre e hijo y este se asocia al id del objeto de aprendizaje*

```
update_metadato_oa_table()
```

**C** \ \ User\_standard\_model

Esta clase contiene los metodos de insercion actualizacion y borrado de metadatos pertenecientes al estandar proporcionado por el usuario Class User\_standard\_model

Package

Default

**M** Methods

Función que borra la fila de una tabla identificando su consecutivo

```
delete_metadato_oa_table($id, $supfather, $padre)
```

Se revisa que exista al menos una fila en la tabla

```
exists_insert_oa($oa) : boolean
```

Esta función busca si existe registrado al menos el oa correspondiente a una tabla

```
exists_oa_table($suppadre, $padre, $oa) : boolean
```

Retorna el consecutivo inmediatamente siguiente al ultimo id del oa que se tiene

```
get_last_row() : integer
```

Esta función devuelve el metadato categoria resultado de la búsqueda en el estandar del usuario

```
get_metadato_father($suppadre) : mixed
```

Funcion que obtiene el parent id del padre

```
get_parentid($padre) : mixed
```

Función que busca y retorna la estructura el estandar de metadatos proporcionado por el usuario

```
get_user_standard() : mixed
```

En caso de no existir fila con el consecutivo del oa se insertara los datos

```
insert_metadato_oa_suppadre($padre, $hijo, $dato)
```

Esta función inserta datos en una tabla determinada, identificandola con los parametros de superpadre, padre e hijo y este se asocia al id del objeto de

## aprendizaje

---

```
insert_metadato_oa_table($supfather, $padre, $hijo, $dato, $oa)
```

## En caso de que exista algun dato se actualizara la fila

---

```
update_metadato_oa_suppadre($padre, $hijo, $dato, $oa)
```

Esta función actualizara datos en una tabla determinada, identificandola con los parametros de superpadre, padre e hijo y este se asocia al id del objeto de aprendizaje

---

```
update_metadato_oa_table($id, $supfather, $padre, $hijo, $dato)
```