

Yatta

Martín Marrese <[mmarre@mecon.gov.ar](mailto:mmarre@mecon.gov.ar)>

MEcon

# Índice

<b>1. Agregar un Proceso</b>	<b>3</b>
1.1. Formato del array de datos . . . . .	3
<b>2. Formato del Archivo Proceso</b>	<b>3</b>

## 1. Agregar un Proceso

El método estático `YATTA_Controlador::agregarProceso` es el encargado de dar de alta los nuevos procesos, y de ponerlos en la cola de ejecución. Este método recibe como parametro una conexión a la base de datos y un array asociativo con los atributos del proceso. Devuelve un `PEAR_Error` en caso de haberlo.

### 1.1. Formato del array de datos

El array asociativo de datos esta compuesto por claves obligatorias y claves opcionales. Es de suma importancia que se respete el formato de la clave (mayusculas/minusculas).

- Obligatorias

1. `script` : path completo del script. Con esto se hara el require once al momento de la ejecución.
2. `id_sistema` : es el identificador numerico asignado al sistema en SAMURAI.
3. `descripcion` : es un texto descriptivo del proceso que será mostrado en la interfaz web al usuario.
4. `owner` : `login@organismo` del usuario que esta iniciando el proceso. Si el proceso genera un archivo resultado, este ocupara espacio en la cuenta de este usuario.

- Opcionales

1. `destinos` : `login@organismo` de los destinos. Son aquellos que tendran acceso al resultado del proceso.
2. `prioridad` : indica la prioridad del proceso. Se utilizara para ponderar la ejecución de los sistemas.
3. `notificar` : indica si hay que avisarle al owner (y destinos) sobre la finalización de la ejecución del proceso.
4. `resultado` : este es el nombre del archivo resultado que hay que mostrarle al usuario en la interfaz web. No es necesariamente el mismo nombre que tendra el archivo real de resultado.

## 2. Formato del Archivo Proceso

Cuando se quiere lanzar el proceso el script que corre en el servidor llama a una función que devuelve un instancia del objeto. Este objeto tiene que cumplir ciertas condiciones para que pueda ejecutarse y obtener los datos cuando finaliza.

La función debe llamarse `create_process`, la cual no recibe parametro alguno. Esta devuelve una instancia del objeto proceso.

El objeto debe tener un metodo `run`, que no recibe parametros, el cual es invocado a la hora de ejecutar el proceso. Una vez que este finaliza debe dejar en 3 atributos que son utilizados para cargar los datos de finalización.

- `error` : contiene cualquier mensaje de error que se hubiera producido.
- `archivo` : contiene el path completo del archivo que se genero como resultado.
- `notificar` : indica si hay que notificar al responsable.