

## Integración con Kongregate API en Sunset Revenge

### Relevamiento:

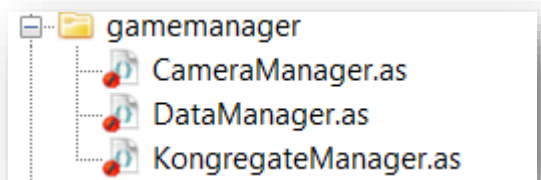
Previamente, se especifica que la plataforma objetivo del proyecto Sunset Revenge es [Kongregate](#) y se tiene por requerimiento incluir un registro de los puntajes más altos obtenidos por los jugadores. A continuación, se procede a detallar el resultado de la etapa de investigación para el desarrollo de la solución mencionada.

1. Kongregate dispone de un portal propio <https://developers.kongregate.com> que ubica los principales sitios de interés a los desarrolladores de videojuegos.
2. Kongregate dispone de un conjunto de herramientas destinada a creadores de videojuegos. Para la presente integración se empleó la API Kongregate. (<https://docs.kongregate.com/docs/client-api-introduction>).
3. El API cuenta con soporte para ActionScript 2 y ActionScript 3, dado que para el desarrollo del videojuego se utilizó la librería Flashpunk se implementa con la opción de AS3. (<https://docs.kongregate.com/docs/actionscript-3-api>).
4. El API brinda distintos servicios. En la implementación se utilizan los servicios de estadísticas (<https://docs.kongregate.com/docs/concepts-statistics>) y de información de usuario. (<https://docs.kongregate.com/docs/concepts-user-information-and-friends>).

### Implementación

La integración en el proyecto se llevó a cabo de la siguiente manera:

1. El código fuente se encuentra segmentado por funcionalidad. Los Manager son clase que permiten administrar componentes del gameplay comunes a todos los mundos de juegos. Cuentan con funciones estáticas que permiten administrar eventos puntuales. El KongregateManager determina el comportamiento ante peticiones de empleo de la API de Kongregate.
2. Para el desarrollo de la clase se utilizó de referencia la documentación de integración en AS3 (<https://docs.kongregate.com/docs/actionscript-3-api>). La



primera restricción identificada es la necesidad de emplear una instancia de un objeto DisplayObject para contar con acceso a las FlashVar. En consecuencia, se optó por incluir la llamada inicial de la función en el Main, para contar con el acceso.

```
_loader = KongregateManager.GetLoaderUser(root.loaderInfo);  
this.addChild(_loader);
```

3. La función GetLoadeUser retorna un Loader configurado, a emplear en los request al API Kongregate. Su referencia se almacena en el Main para un empleo posterior.

```
public static function GetLoaderUser(Info:Object):Loader  
{  
    paramObj = LoaderInfo(Info).parameters;  
    apiPath = paramObj.kongregate_api_path || "http://cdn1.kongregate.com/flash/API_AS3_Local.swf";  
    Security.allowDomain(apiPath);  
    Security.allowInsecureDomain(apiPath);  
    request = new URLRequest(apiPath);  
    loader = new Loader();  
    loader.contentLoaderInfo.addEventListener(Event.COMPLETE, get_user_data);  
    loader.load(request);  
    return loader;  
}
```

4. La función callback get\_user\_data almacena el username y ID del usuario si es posible, cuando la petición es atendida.

```
public static function get_user_data(event:Event):void  
{  
    kongregate = event.target.content;  
    kongregate.services.connect();  
    if (!kongregate.services.isGuest()) {  
        DataManager.set_username(kongregate.services.getUsername());  
        DataManager.set_iduser(kongregate.services.getUserId());  
    }else {  
        trace("Usuario Invitado - No se pueden obtener datos del usuario")  
    }  
}
```

5. Al finalizar los niveles y pasar a la vista de créditos se efectúa la petición al Main de la función SaveScore, que tras la llamada anidada a SetLoaderScore, solicita el registro del puntaje más alto.

```
public static function SetLoaderScore(l:Loader):void {
    l.contentLoaderInfo.addEventListener(Event.COMPLETE, set_user_score);
    l.load(request);
}

public static function set_user_score(event:Event):void
{
    kongregate = event.target.content;
    kongregate.services.connect();
    if (!kongregate.services.isGuest()) {
        kongregate.stats.submit("HighScore", DataManager.getScore());
    }else {
        trace("Usuario Invitado - No se puede registrar las estadísticas")
        //kongregate.stats.submit("HighScore", DataManager.getScore()); For Shadow Valid
    }
}
```