

GamesAPI

API de videojuegos

Valentín Castravete – 3º DAM DAW DUAL

Índice

Enlaces de acceso al contenido y despliegue del mismo.....	3
Introducción.....	4
Objetivos	6
Análisis.....	7

Enlaces de acceso al contenido y despliegue del mismo

GitHub: <https://github.com/valentincastravete/GamesAPI>

Introducción

Este proyecto trata sobre una **API REST pública y abierta de videojuegos**.

API

Una **API** es un conjunto de código de programación que permite la transmisión de datos entre un producto de software y otro. También contiene los términos de este intercambio de datos.

Tipos de API

En términos de disponibilidad, las API pueden ser privadas, de socios y públicas.

Privadas: Estas interfaces de software de aplicación están diseñadas para mejorar las soluciones y servicios dentro de una organización. Los desarrolladores o contratistas internos pueden usar estas API para integrar los sistemas de TI de una empresa, integrar aplicaciones, crear nuevos sistemas o aplicaciones orientadas al cliente que aprovechen los sistemas existentes.

De socios: Las API de socios se promocionan abiertamente, pero se comparten con los socios comerciales que han firmado un acuerdo con el editor. El caso de uso común para las API de socios es la integración de software entre dos partes.

Pública: También conocidas como orientadas al desarrollador o externas, estas API están disponibles para cualquier desarrollador de terceros. Un programa de API pública permite aumentar el conocimiento de la marca y recibir una fuente adicional de ingresos cuando se ejecuta correctamente.

Hay **dos tipos de API públicas: abiertas** (gratuitas) y **comerciales**. La definición de API abierta sugiere que todas las características de dicha API son públicas y se pueden usar sin términos y condiciones restrictivos. Por ejemplo, es posible crear una aplicación que utilice la API sin la aprobación explícita del proveedor de la API o las tarifas de licencia obligatorias. La definición también establece que la descripción de la API y cualquier documentación relacionada deben estar disponibles abiertamente y que la API se puede utilizar libremente para crear y probar aplicaciones.

Hay diferentes **tipos de usos para las API**, como la API **remota**, de **sistemas operativos, web y bases de datos**.

El de este proyecto está orientado a una **API de base de datos**, que permiten la comunicación entre una aplicación y un sistema de administración de bases de datos. Los desarrolladores trabajan con bases de datos escribiendo consultas para acceder a datos, cambiar tablas, etc.

Y según el **protocolo** que usa, es **REST**, es decir, cada recurso está representado por una URL única, y se puede solicitar este recurso proporcionando su URL.

La información devuelta se encuentra en JSON, que contiene colecciones de pares nombre/valor y listas ordenadas de valores. Dado que se trata de estructuras de datos universales, el formato se puede utilizar con cualquier lenguaje de programación.

Videojuegos

Los **videojuegos** son juegos digitales, jugados en plataformas como un ordenador, consolas, móviles, etc. Estos requieren de una interfaz de usuario y/o un periférico de entrada como un mando, teclado, etc., para generar una respuesta. Esta respuesta es mostrada en un dispositivo de visualización como una televisión, monitor, gafas de realidad virtual, etc. Algunos de estos videojuegos incluyen sonidos y/o música para mejorar la experiencia de usuario.

Los videojuegos se pueden **clasificar gracias en base a estas características**:

Género: Es la principal característica que define un videojuego. Define el estilo del mismo en base al modo de juego. Algunos de estos son juegos de disparos, de terror, fantasía, medieval, etc.

Modo: Esto especifica cuántos jugadores pueden jugar, en qué momentos, desde qué perspectivas y el orden entre todo ello. Los principales son un solo jugador y multijugador.

Intenciones: Indica la orientación, finalidad, emociones o sensaciones a generar en el usuario final definidas por la forma de jugar el mismo, aunque bajo mi punto de vista, esto sigue dependiendo en una pequeña parte de como el usuario final quiera jugar el videojuego, pero sin esto cambiar las intenciones iniciales del creador del videojuego.

En el caso del **primer videojuego del que va a ofrecer información mi proyecto** es **Monster Hunter World**.

Es un videojuego con **género de acción y mundo abierto, modo multijugador y cooperativo**. Pretende unir tanto la gran historia que tiene detrás como la gran y amplia acción que ofrece, además del juego entre otros jugadores que lo hace más enriquecedor.

Intenciones

La información que ofrece mi API está orientada a las personas y/u organizaciones que quieran realizar una **wiki** acerca del videojuego, es decir, maquetarla para ofrecerla de una forma visual al cliente final, que serían los jugadores de los propios videojuegos, los cuáles acudirían a esta por ayuda o intereses personales acerca del videojuego.

Aparte de la propia API, también habrá una página web que ofrecerá, mediante un fácil filtrado, el enlace por el cual hay que pedir a la API una información en concreto. Este filtrado sería variable en base al videojuego del que se ofrece la información.

Una **wiki** es una página web en la cual se documenta en profundidad cualquier dato, información, forma de jugar, etc., acerca de un tema en concreto (en nuestro caso para los videojuegos).

Objetivos

Estos son todos los objetivos que busco obtener al realizar este proyecto:

- Ofrecer una API REST pública y abierta de uno o varios videojuegos.
- Mantener los datos ofrecidos actualizados periódicamente.
- Facilitar a las personas o entidades que hagan uso de mi API la creación de documentación acerca de uno y/o varios videojuegos.
- Ofrecer una página web que ofrezca URLs a la API diferentes en base a una configuración seleccionada por el usuario final.
- Aprender a desarrollar una API con una librería de php orientada exclusivamente a este tipo de API.
- Realizar un correcto desarrollo y mejorar en todos los aspectos relacionados a programar, sobre todo orientado a videojuegos, ya que es un campo que personalmente me gusta.
- Finalizar el curso actual para poder proseguir con mi desarrollo personal, no profesional.

Análisis

Proyectos similares

Tras buscar detenidamente varias API que ofrezcan este mismo tipo de información, me he encontrado con este [proyecto](#) que me ha llamado la atención, llamado mhw-db, el cuál ofrece exactamente la misma información que el proyecto que estoy realizando.

Según mi búsqueda, este ha sido el mejor proyecto con una misma finalidad que el mío. El resto de proyectos encontrados son proyectos incompletos, falta de información o no internacionalización.

Pero la diferencia principal entre este proyecto y el mío, es que mhw-db ofrece las URL de información explicadas en una web con documentación de ello, pero no expone todas las URL disponibles. Lo que mi proyecto si tendrá será una página web principal con unos filtros y desplegables indicando la URL de los recursos o recurso que el usuario esté buscando, facilitándole la búsqueda.

Tras buscar de donde ofrecía este proyecto su información, he encontrado este proyecto de github con información estructurada del videojuego:

[MHWorldData](#)

El cuál ofrece en archivos csv, un archivo parecido a un archivo Excel, la información de casi todo el videojuego.

mhw-db se dedica a leer estos archivos csv y crear su propia base de datos.

Lo que mi proyecto quiero que haga es que, he buscado y parece que el proyecto de MHWorldData ofrece uno o varios archivos para crear la base de datos directamente, lo cual ahorraría un paso importante.

Modo de desarrollo y herramientas a utilizar

Al comenzar con todo esto, lo primero en lo que pensé fue en usar un framework conocido como Symfony o Laravel, orientados a desarrollo de proyectos con diferentes orientaciones, entre estas, una API.

Pero tras leer y analizar, me di cuenta de que ninguna de las dos se adaptaba del todo al propósito y tamaño de mi proyecto. Ya que Symfony está orientado a proyectos muy grandes, como PrestaShop, y que Laravel, aunque sí que está

orientado a proyectos más pequeños, sigue siendo muy complejo para la sencillez que yo busco en mi proyecto.

Gracias a uno de los jefes de la empresa en la que estoy cursando mi formación en centro de trabajo, que me ha recomendado una librería de puro php que facilita mucho la creación de una API, con simples funciones por llamada o URL, sin definir ningún controlador y/o modelo. Únicamente hay que definir las rutas y el archivo de acceso a la bd.