

## Autores:

HIMAR MANUEL BARQUÍN CARRASCO

CARLOS PÍO REYES

## Localización y explicación de los requisitos:

- **XML:**

El XML que se usa en la página se encuentra en la carpeta xml y se llama data.xml

- Sintaxis correcta: Utilizamos una sintaxis sencilla e intentando que esté lo más organizada posible.
- Longitud de datos suficiente: Guardamos con XML la información de los juegos que tiene disponible nuestra página. En concreto, almacenamos la información de 8 juegos: su título, fecha de publicación, distribuidor, desarrollador, entre otros. Mirar en el fichero 'data.xml' en la carpeta xml para más información.
- Inclusión de atributos: Utilizamos atributos para cada etiqueta 'game' y 'picture'.

- **XSL:**

Los ficheros XSL se encuentran en la carpeta xml del proyecto.

- Generación del fichero HTML a partir de la plantilla: Generamos en concreto trozos de código (cuerpo de la página, sin el header ni el footer), de 9 páginas HTML. La información de cada juego y la página donde se ven todos los juegos.

- **HTML:**

- Estructura página web: Los ficheros HTML se encuentran en la raíz del proyecto. Los ficheros CSS y JS se encuentran en sus carpetas correspondientes. Además, disponemos de las carpetas 'docs' e 'img' para documentación e imágenes respectivamente.
- Elementos cabecera HTML: En la etiqueta 'head' de cada fichero HTML se encuentran todos los datos solicitados (título, icono personalizado, etiquetas meta, etc).
- Elementos cuerpo HTML:

- **Título del proyecto:** Se encuentra en el header de la página. El título de nuestra página es 'TrophieGames'.
- **Mensaje principal:** En el fichero 'index.html' se encuentra una breve descripción del objetivo de la página.
- **Imágenes identificativas del proyecto:** En la página principal, podemos ver lo denominado carousel (desplazamiento con imágenes). Ahí se puede observar diferentes imágenes del mundo de los videojuegos.
- **Formulario de registro:** En la parte superior derecha de las páginas se pueden encontrar enlaces para iniciar sesión y para registrarse. En el último caso disponemos de un formulario con validación y que solicita tanto nombre de usuario como correo y contraseña, además de ciertos datos propios de la página web.
- **Validación de los formularios:** Se comprueba correctamente los datos introducidos, así como que los campos estén vacíos, en el formulario de registro.
  
- Para visualizar las diferentes guías de la página, así como los foros, se utiliza un sistema basado en la URL. Enviamos mediante la URL al fichero 'foro.html' y 'guia.html' el identificador de la guía o el foro que queremos observar. Por ejemplo, si quisiéramos observar la guía con ID 3 la URL sería la siguiente:

**<https://alu0101119373.github.io/trophiegames/guia.html?id=3>**

- Uso de HTML5 para control del formulario: En el formulario de registro, el de login y el de perfil (al solicitar ciertos cambios en el perfil, como el nombre de usuario) podemos observar ciertos elementos introducidos en HTML5.
  
- **CSS + Javascript:**
  - Imagen o imágenes de fondo: Hemos utilizado una imagen de fondo para el banner, que se encuentra en todas las páginas de nuestro proyecto. No hemos considerado necesario utilizar clases de CSS para imágenes porque con la etiqueta img teníamos suficiente. El código que da la imagen de background al banner se encuentra en la carpeta css, fichero styles.css líneas de la 209 a la 214 del código en la clase banner.
  - Colores, fuentes de texto, aspectos visuales personalizados: carpeta css, fichero styles.css casi todas las clases modifican algunos de los aspectos arriba mencionados.
  - Comprobar el formato del email: En la carpeta js, fichero login.js, línea del código 26 podemos observar la validación del email, con los requisitos solicitados para el proyecto (que tenga todos los campos del email un mínimo

de 2 caracteres).

- Otros elementos JavaScript: carpeta js, analisis.js, crear-guia.js, crear-foro.js, events.js, firebase.js, foro.js, foros.js, guias.js, isLoggedIn.js, login.js, signin.js implementan otras funcionalidades de JavaScript. Casi todo el proyecto se basa en JavaScript y en Firebase para mostrar los datos de la base de datos.
- Bootstrap:
  - Definición del grid con bootstrap para organizar la página: en la mayoría de ficheros HTML al inicio de la etiqueta body ya se empieza a definir el grid para organizar la página. En prácticamente todas las páginas se utilizan atributos del grid como las diferentes clases 'col' y el uso de row. En la sección de juegos se utiliza grid para mostrar los diferentes juegos disponibles.
  - Visualización de la página correcta en diferentes dispositivos y resoluciones: en la mayoría de ficheros HTML al definir el grid se hace también la definición del mismo para diferentes dispositivos, además del uso de varias clases de css del fichero styles.css, en la carpeta css para conseguir este objetivo. La página está adaptada a diferentes dispositivos, utilizando tanto las clases que ofrece Bootstrap como el uso de '@media' de CSS.  
Por desgracia, no nos dio tiempo a hacer responsive los foros.
  - Uso de otros elementos bootstrap: en el header de la página y cuando el usuario está iniciado (para acceder al perfil y al botón de cierre de sesión) se hace uso de un dropdown, además de que en el fichero index.html se hace uso de un carousel para mostrar imágenes, varias inputs forms y buttons en la creación de una guía y en la página de registro e inicio de sesión. Se usa también objetos Modal de Bootstrap para la configuración del perfil. Elementos de formularios como el textarea, los buttons, los input text y similares (password, email, file, etc) también se utilizan en todos los formularios que dispone la página (login y signup, creación de foro, creación de guía, configuración del perfil, etc).
- Firebase:
  - Autenticación de usuarios: Los usuarios se pueden tanto registrar como iniciar sesión. En ambos casos, tienen dos opciones: o mediante correo y contraseña o mediante el proveedor de Google. También pueden modificar los datos de su perfil (en la pestaña perfil del usuario) así como borrarse la cuenta.

- Analytics. Añadir capturas:



- Otro/s servicio de Firebase, BBDD, Ads, Crashlytics, etc : Utilizamos el Storage de Firebase y la Base de Datos para manipular la información que maneja la página. Todos los trofeos de cada juego, información básica de cada uno de ellos, así como todas las guías, foros, valoraciones de los usuarios, enlaces a análisis, datos adicionales de los usuarios (imagen de perfil, nombre de usuario, mensaje personal, etc). Storage lo usamos para almacenar las imágenes de los usuarios para obtener luego sus URL y poder ser visualizadas cuando se requiera.