Patricia Tovar Quintano

IES GALILEO

17/05/2022

inook

Contenido

[INTRODUCCION 3](#_Toc104845541)

[DOCUMENTACION 3](#_Toc104845542)

[Objetivo 3](#_Toc104845543)

[Temática 3](#_Toc104845544)

[MANUALES 3](#_Toc104845545)

[Manual del Código 3](#_Toc104845546)

[Estructura de todo el código 3](#_Toc104845547)

[Acceso a Datos 3](#_Toc104845548)

[Descripción Grafica 4](#_Toc104845549)

[Excepciones que hemos tenido en cuenta: 5](#_Toc104845550)

[Manual del Usuario: 6](#_Toc104845551)

[Instalación de XAMPP y ejecutar el servidor 6](#_Toc104845552)

[Página Web: 6](#_Toc104845553)

[CONCLUSIONES 8](#_Toc104845554)

[WEBGRAFIA 9](#_Toc104845555)

[Herramientas de Software (Recursos) 9](#_Toc104845556)

[Investigación Librerías 9](#_Toc104845557)

[JQUERY 9](#_Toc104845558)

[Diseños Seguidos 9](#_Toc104845559)

[JSON 9](#_Toc104845560)

[Tabla – Librería Adicional para presentar Datos 9](#_Toc104845561)

[Acceso a Base de Datos 9](#_Toc104845562)

[API 9](#_Toc104845563)

[XMLHttpRequest + XML + GET. 9](#_Toc104845564)

[XMLHttpRequest + JSON + GET. 9](#_Toc104845565)

[API Fetch + XML + GET 10](#_Toc104845566)

[API Fetch + JSON + GET. 10](#_Toc104845567)

# INTRODUCCION

Mi proyecto va a tratar de una aplicación que realiza conexiones a archivos externos al código, en nuestro caso para conseguirlos usaremos los de una API semipública, Esta se puede usar, pero hay que solicitar un código de acceso porque ellos llevan un control de quien la usa y para qué.

Además, también usaremos unas conexiones usadas desde un servidor de XAMPP.

# DOCUMENTACION

## Objetivo

La aplicación tiene el fin de ser Informativa y la idea es entregar una aplicación en desarrollo, ya que por tiempo no nos dará tiempo a tocar todas las variables de dicha aplicación.

## Temática

Nuestra temática es el Videojuego *Animal Crossing New Horizons* (ACNH).

Al enfocarnos en hacer una aplicación web informativa sobre ACNH el nombre va a ser iNook, *Nook* hace referencia a uno de los personajes principales TOM NOOK, un mapache que va guiando el juego y quien dirige la isla en la de cada jugador.

# MANUALES

## Manual del Código

### Estructura de todo el código

Cada página tendrá sus propios ficheros de estilos CSS y JS, además el proyecto cuenta con un directorio de datos donde están los ficheros XML y JSON donde están los datos para poder hacer el POST y algunos GETTERS.

También está el menú izquierdo “S*idebar*” que tiene su propio JS y CSS y también hemos separado el código general, que usa cualquier otro fichero, en sus respectivos ficheros de JS y CSS.

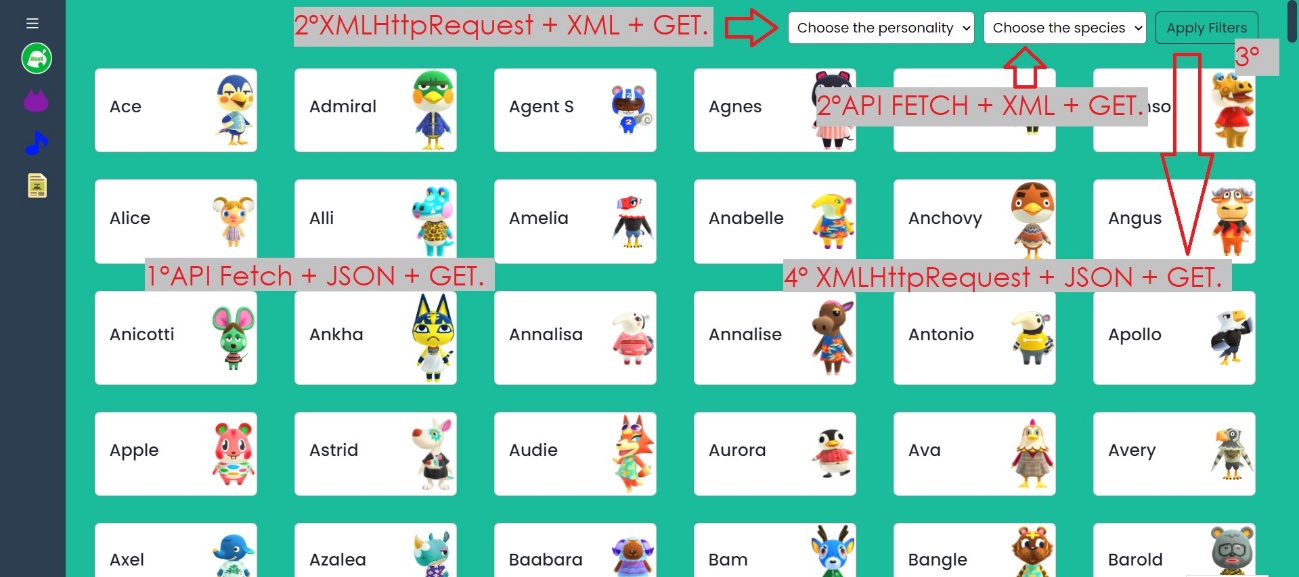
La estructura en los Fichero de JS 🡪 Funciones (Ordenadas alfabéticamente), variables, ejecución de código y eventos al final del fichero.

### Acceso a Datos

* **XMLHttpRequest + XML + GET.** 🡪 Aplicado en [Aldeanos](#_Aldeanos) (MostrarSelectPersonalidad)
* **XMLHttpRequest + XML + POST.** 🡪 Aplicado en [Colecciones](#_Colecciones) (CrearSelectMes)
* **XMLHttpRequest + JSON + GET.** 🡪Aplicado en [Aldeanos](#_Aldeanos) (AplicarFiltros)
* **XMLHttpRequest + JSON + POST.** 🡪 Aplicado en la [Música](#_Música) (Mostrar Canciones)
* **API Fetch + XML + GET.** 🡪 Aplicado en [Aldeanos](#_Aldeanos) (MostrarSelectEspecies)
* **API Fetch + XML + POST.** 🡪 Aplicamos en [Colecciones](#_Colecciones) (CrearBotonesHemisferio)
* **API Fetch + JSON + GET.** 🡪 Aplicado en [Aldeanos](#_Aldeanos) (MostrarTodos) y en [Colecciones](#_Colecciones) (MostrarCapturopedia)
* **API Fetch + JSON + POST**. 🡪 Aplicado en [Colecciones](#_Colecciones) (CrearBotonesCriaturas)

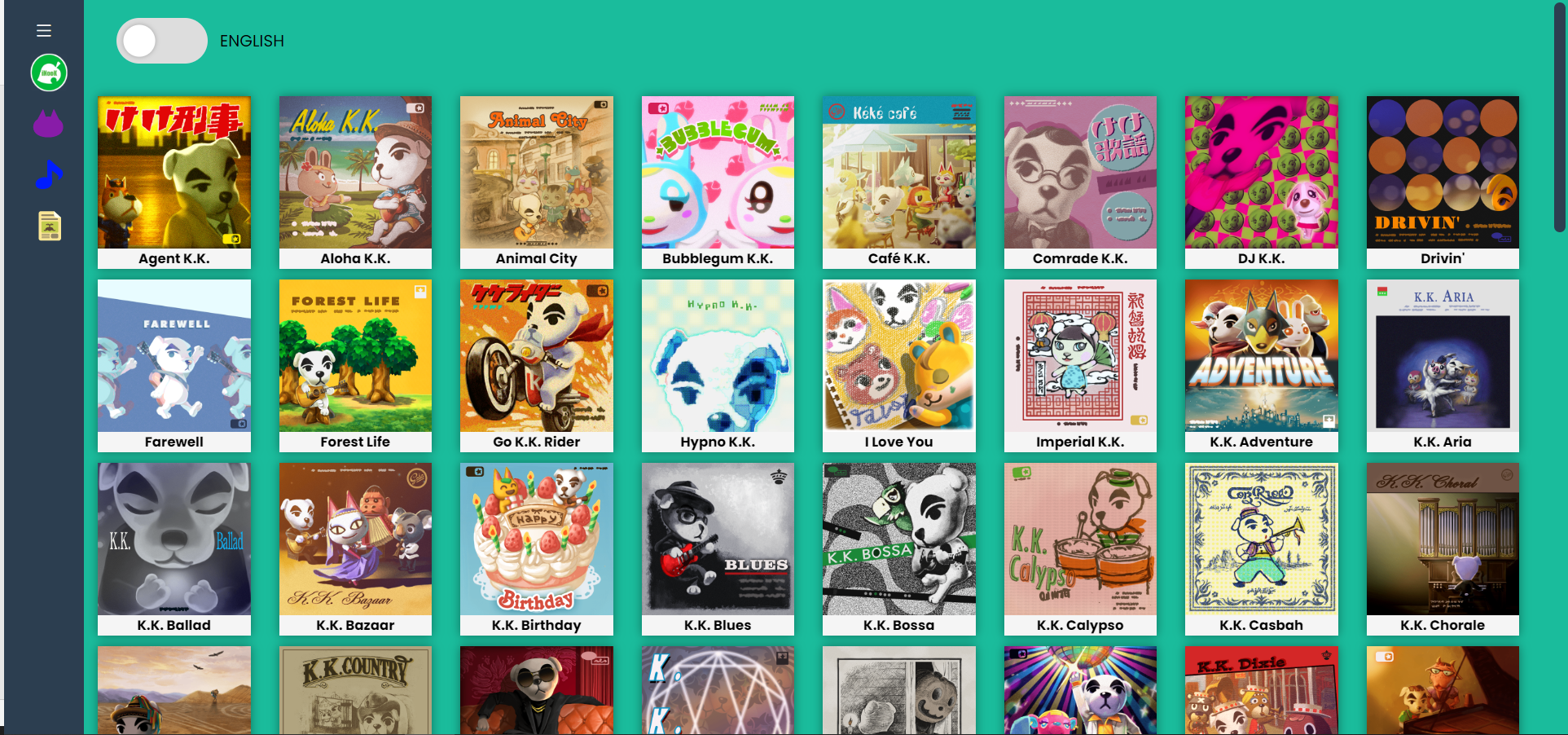
### Descripción Grafica

#### Aldeanos



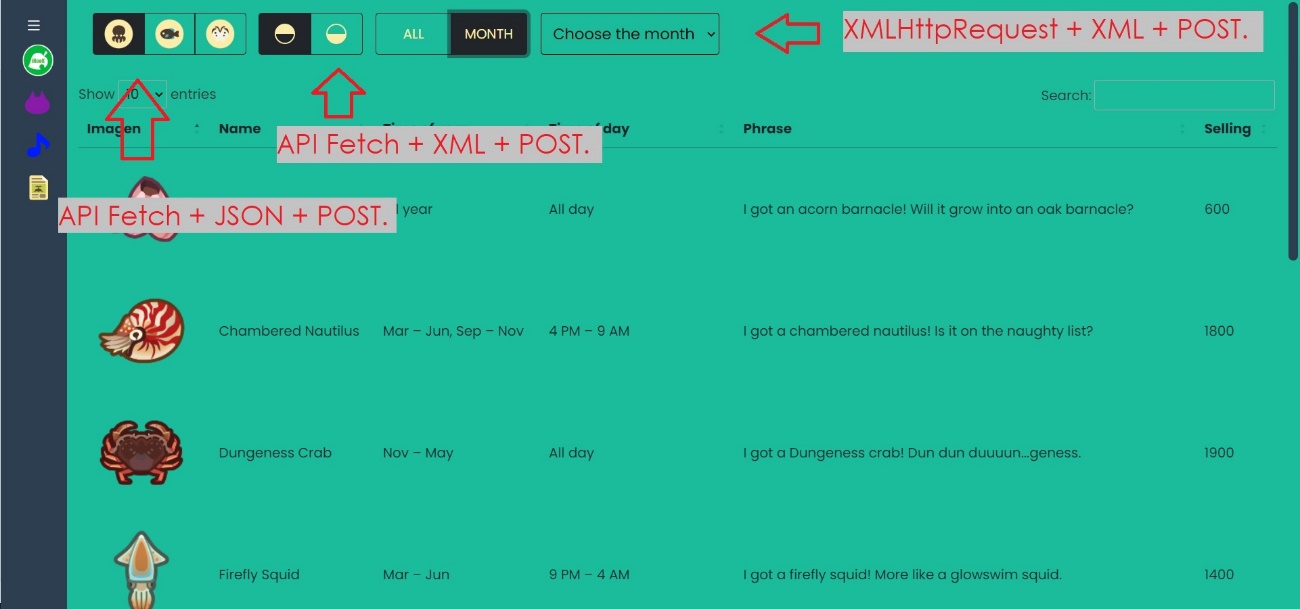
1. Generamos con **Api Fetch + Json + Get** todos las Tarjetas de los aldeanos.
2. Generamos el Select de la personalidad (**XMLHttpRequest + XML + GET**.) y las Especies (**API Fetch + XML + GET**.).
3. Al hacer Click en Aplicar Filtros generamos el evento
4. Se muestran los aldeanos que tienen el filtro (**API Fetch + JSON + GET**).

#### Música



Generamos la música con **XMLHttpRequest + JSON + POST**, lo que son las álbumes de música.

#### Colecciones



1. Hemos generado los botones con los Post restantes (como se indican en la foto).
2. Imprimimos las criaturas seleccionadas con **API Fetch + JSON + GET**, como hemos usado en aldeanos.

### Excepciones que hemos tenido en cuenta:

Permitir que los filtros puedan coexistir, es decir, que uno no anule a otro.

En caso de que no se encuentren datos al aplicar unos filtros se mostrará un mensaje informativo ‘*no se ha encontrado la información solicitada*’. Por ejemplo: En la pestaña de *Villagers* [[1]](#footnote-1) si ponemos en la personalidad *Smug*[[2]](#footnote-2) y en especie ‘*cub*[[3]](#footnote-3)’ podemos ver dicho mensaje.

Tener que dar a Play cuando se haga clic en la portada de la canción y que no se reproduzca automáticamente

Los coleccionables tienen unos filtros por defecto, tendrá elegidos el primer hijo.

No siempre se aplicará el filtro de meses, hay que diferenciar cuando aplicamos meses y cuando no.

Por defecto si no se aplica un mes, coge el actual (valor de la variable current).

## Manual del Usuario:

### Instalación de XAMPP y ejecutar el servidor

Usaremos el paquete de software libre [**XAMPP**](https://www.apachefriends.org/es/index.html) que sirve para todas las plataformas, también esta LAMPP para Linux… por eso es mejor usar XAMPP, este paquete tiene varias aplicaciones, la que nos interesa es **Apache,** servidor Web HTTP.

Para levantar el servidor solo tendremos que levantar el servidor dando a la tecla **Start** y tiene que quedar como se ve en la imagen.



Nuestro Proyecto pegaremos en la ruta **C:\xampp\htdocs** ejemplo:

C:\xampp\htdocs\iNook Patricia\

Y, por último, pegaremos dicha ruta en la URL de nuestro navegador y **sustituiremos desde c hasta htdocs por localhost** en nuestro ejemplo quedaría como, por ejemplo: localhost\iNook Patricia

Nota: como nuestro proyecto hemos puesto que el fichero principal índex.hmtl no hace falta que indiquemos este fichero, le ejecutara primero directamente si no hubiese ningún índex, indicaríamos el fichero que queremos ejecutar ejemplo: localhost\iNook Patricia\index.html

### Página Web:

He decidido que todas tengan un menú vertical desplegable en la parte izquierda para poder ir entre pagina y pagina, dejando el resto de la pantalla para representar los datos, en la zona de datos en la parte superior están situados un encabezado donde suelen estar los filtros.

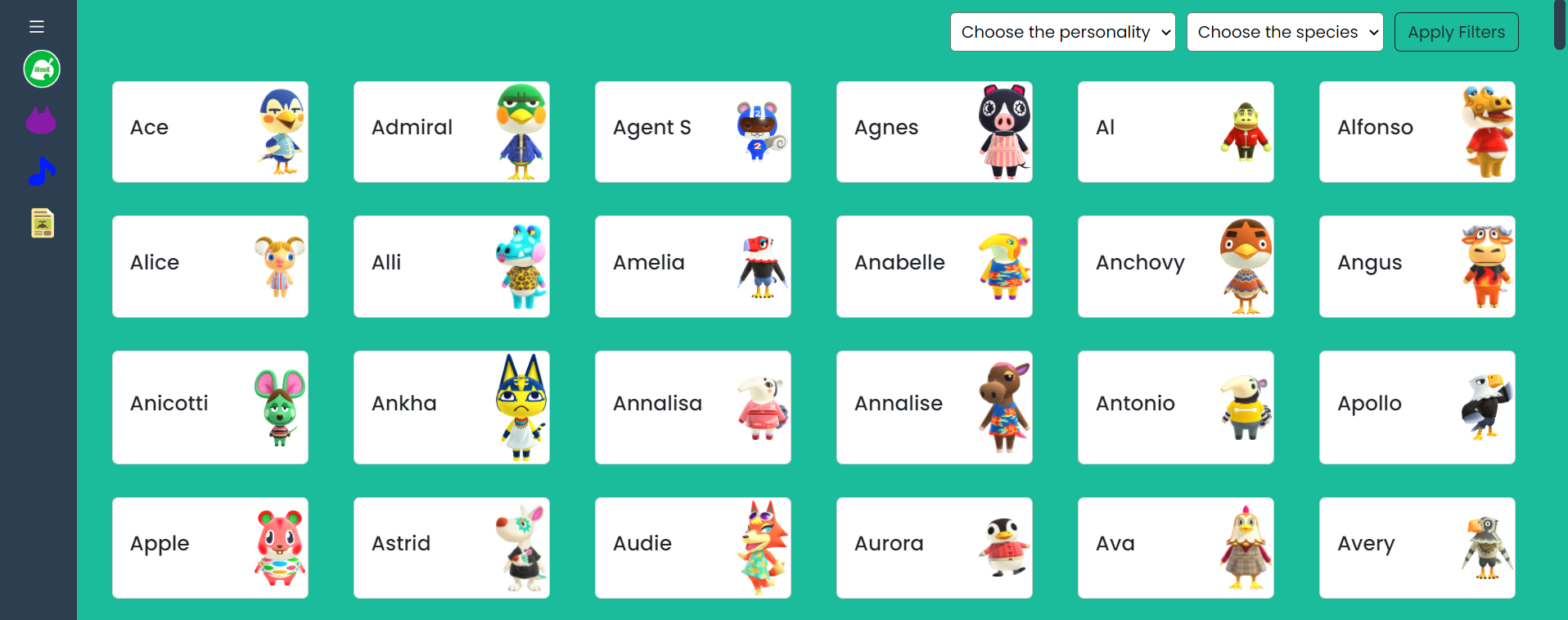
#### Menú (Home)

Aparece un menú de con los distintos apartados que hemos creado:

Villagers, Music, Collectable[[4]](#footnote-4)

#### Aldeanos (Villagers)

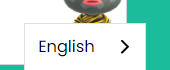
De forma inicial se mostrarán todos los Aldeanos.



En la parte superior están los filtros de personalidad y especies.

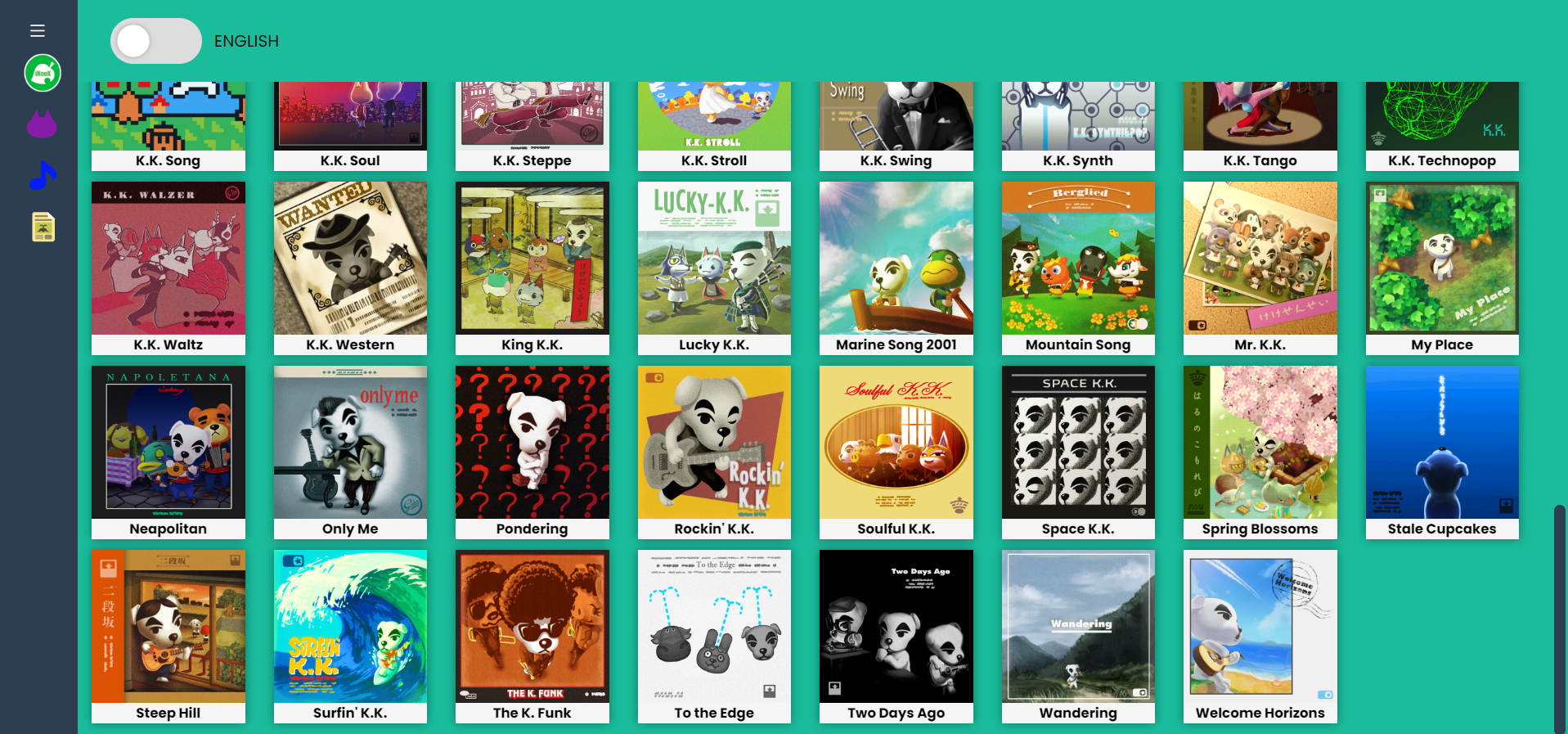
Para aplicar los filtros solo tendremos que seleccionarlos y pulsar el botón de aplicar filtros, cuando se selecciona un filtro aparece a la derecha un botón de borrar filtro, cuando este se seleccione se quita la selección, pero hay que tener en cuenta que hay que volver a pulsar Aplicar Filtros (Apply Filters)

Esta página cuenta también con una librería que traduce la página en español.



#### Música

En esta pagina se encuentran todas las canciones del juego.

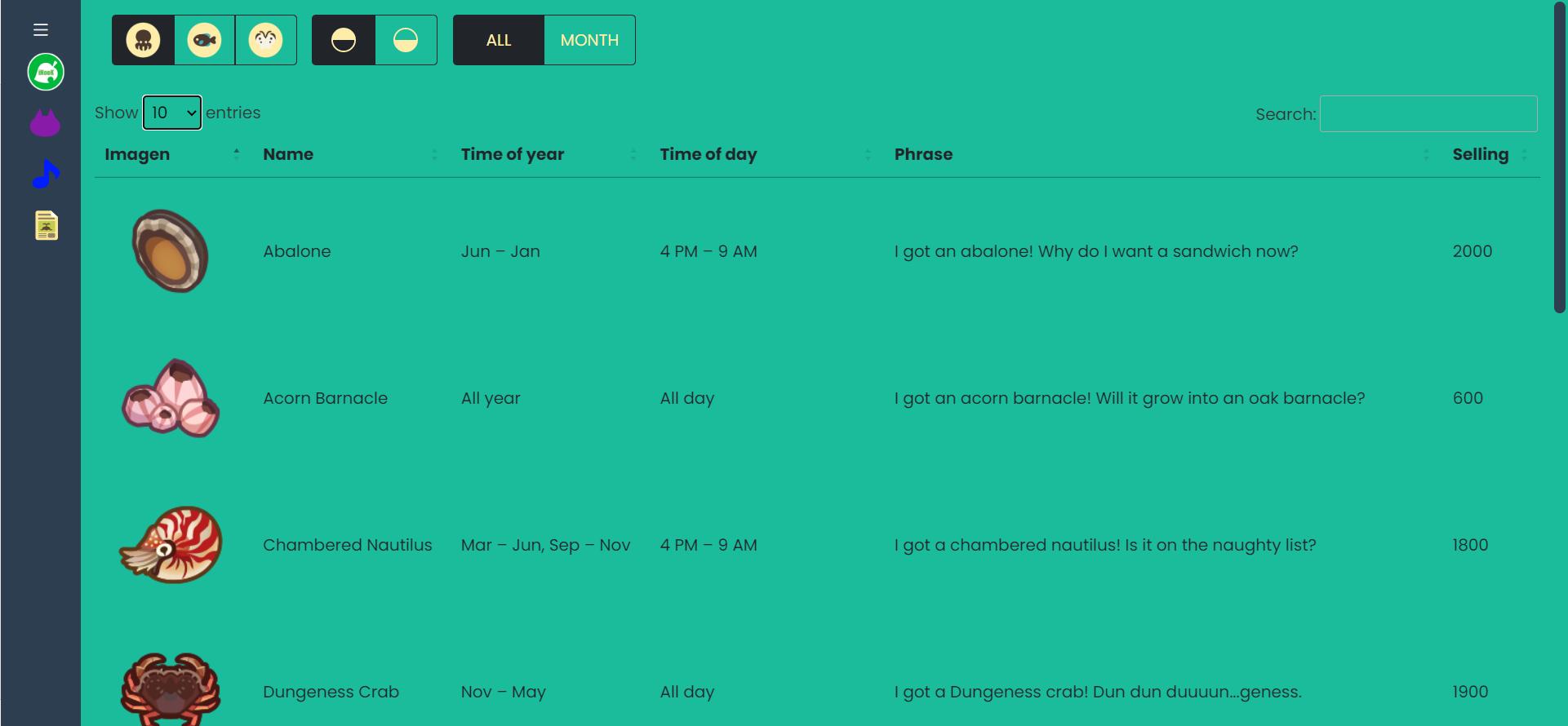


En la parte superior se encuentra El filtro que cambia los títulos de español a ingles y muestra el audio de la canción cuando seleccionamos una canción.



#### Coleccionables

Esta página muestra las criaturas coleccionables: Criaturas de Mar, Peces e Insectos.



Cuando se carga la pagina se muestra las imágenes con los filtros seleccionados por defecto aplicados.

Los datos se muestran en una tabla donde se ven la imagen, el nombre de la criatura el mes del año en el que aparecen, la hora, la frase que dice nuestro isleño cuando le pesca y el precio que tienen para venderle.

# CONCLUSIONES

La Página Web no está acabada completamente, pero si que he realizado 3 páginas donde el usuario puede interactuar, los datos los he conseguido mostrar de las 8 formas posibles de conectar a bases de datos externos.

He visto que a mejorar lo que tengo es que muestre más información, por ejemplo, en los Aldeanos, poder seleccionar al Aldeano y que te muestre más información, única y exclusiva de ese Aldeano.

Otra mejora es optimizar la Aplicación para que el tiempo de respuesta se reduzca.

De este proyecto he notado los tres meses que llevo haciendo JS de forma intensiva, me he encontrado con mucha facilidad y rapidez en lógica de programación, sin embargo, he tenido dificultades en ajustarme a la sintaxis con JQuery y a encontrar información en otras API’s.

# WEBGRAFIA

## Herramientas de Software (Recursos)

[Boostrap](https://getbootstrap.com/docs/5.2/getting-started/introduction/) – Generar partes del HTML como los Cards, iconos, botones…

[XAMPP](https://www.apachefriends.org/es/index.html) – Servidor Web

## Investigación Librerías

<https://kinsta.com/es/blog/bibliotecas-javascript/>

<https://fighters.ru/es/text-pagination---jquery-auto-pagination-ili-avtomaticheskaya-paginaciya-teksta-umnye/>

## JQUERY

<https://stackoverflow.com/questions/11179406/jquery-get-value-of-select-onchange>

<https://es.stackoverflow.com/questions/40754/crear-elementos-del-dom-con-jquery>

<https://www.anerbarrena.com/jquery-empty-4564/>

<https://es.acervolima.com/como-seleccionar-el-primer-elemento-en-la-lista-desplegable-usando-jquery/>

<https://learn.jquery.com/using-jquery-core/faq/how-do-i-check-uncheck-a-checkbox-input-or-radio-button/>

## Diseños Seguidos

<https://bootstrapious.com/tutorial/sidebar/index4.html>

[https://mdbootstrap.com/docs/standard/components/cards/#](https://mdbootstrap.com/docs/standard/components/cards/)

## JSON

<https://www.lawebdelprogramador.com/codigo/JQuery/2977-Recorrer-una-variable-json-con-jquery.html>

<https://www.delftstack.com/es/howto/javascript/javascript-append-to-object/>

## Tabla – Librería Adicional para presentar Datos

<https://datatables.net/>

<https://datatables.net/manual/tech-notes/3>

## Acceso a Base de Datos

### API

<https://api.nookipedia.com>

f66873e9-3cf3-4b56-a17b-1987afe8c3de

### XMLHttpRequest + XML + GET.

<https://www.youtube.com/watch?v=XLMrsRGcOyk>

### XMLHttpRequest + JSON + GET.

<https://stackoverflow.com/questions/1973140/parsing-json-from-xmlhttprequest-responsejson>

<https://www.w3schools.com/xml/xml_http.asp>

<https://stackoverflow.com/questions/1973140/parsing-json-from-xmlhttprequest-responsejson>

### API Fetch + XML + GET

<https://codetogo.io/how-to-fetch-xml-in-javascript/>

### API Fetch + JSON + GET.

<https://www.digitalocean.com/community/tutorials/how-to-use-the-javascript-fetch-api-to-get-data-es>

1. Villagers 🡪 Aldeanos en español [↑](#footnote-ref-1)
2. Smug 🡪 Presumido en español [↑](#footnote-ref-2)
3. Cub 🡪 Cachorro en español [↑](#footnote-ref-3)
4. Collectable 🡪 coleccionable, nos referimos a la parte coleccionable de los animales, en juego en español se llama capturopedía, en ingles no lo he encontrado y la palabra que he encontrado para definirlo es esta. [↑](#footnote-ref-4)