

I.E.S Fuengirola N°1

DESARROLLO DE APLICACIONES MULTIPLATAFORMA

PROYECTO INTEGRADO



Departamento de Informática

Manual Técnico

Autor: Eris Barranco Prieto

Índice de contenidos

1- Sobre éste proyecto.....	3
1.1- Control de Versiones.....	3
1.2- Licencia de Uso.....	3
2- Análisis del problema.....	3
2.1- Introducción al problema.....	3
2.2- Antecedentes.....	3
2.3- Objetivos.....	3
2.4- Requisitos.....	4
2.4.1- Funcionales.....	4
2.4.2- No Funcionales.....	4
2.5- Recursos.....	4
2.5.1- Software.....	4
2.5.2- Hardware.....	4
3- Diseño de la solución software.....	5
3.1- Modelados.....	5
3.1.1- Casos de Uso.....	5
3.2- Prototipado Gráfico.....	5
3.2.1- [Escritorio].....	5
3.3- Base de Datos.....	5
4- Implementación.....	6
4.1- Codificación.....	6
4.1.1- Backend.....	6
4.1.2- Frontend.....	6
4.2- [Pruebas].....	6
5- Documentación.....	7
5.1- Empaquetado y distribución.....	7
5.2- Instalación.....	7
6- Conclusiones.....	7
7- Bibliografía.....	8

1- Sobre éste proyecto.

1.1- Control de Versiones

- Este proyecto usa Git para el control de versiones. Y el repositorio está hosteado en [iesfuengirola1es/proyectointegrado2425-erisbarrancop](https://github.com/iesfuengirola1es/proyectointegrado2425-erisbarrancop).

1.2- Licencia de Uso

- Este proyecto está trabajado bajo la licencia ISC. Que permite usar, copiar y modificar y/o distribuir esta aplicación para cualquier propósito, con o sin incentivo.

2- Análisis del problema.

2.1- Introducción al problema.

- En los últimos años, el consumo y creación de la música ha crecido notablemente. Lo que genera que nuevos artistas emergentes vean más complicado hacerse conocer, y a su vez ven una falta de una comunidad que apoye su visión artística personal, lo cual es esencial para el desarrollo del mismo.

2.2- Antecedentes.

- Las plataformas de música actuales, suelen estar bastante colapsadas, y se pueden llegar a sentir bastante competitivas, haciendo que nuevos artistas lo tengan más difícil para destacar.

2.3- Objetivos.

- **Objetivo General:**
 - Crear una plataforma web en Wordpress, para el apoyo de artistas independientes, permitiéndoles compartir sus proyectos y recibir un incentivo a modo de donación de los usuarios.
- **Objetivos Específicos:**
 - Implementar una barra de búsqueda en tiempo real para singles y álbumes, permitiendo al usuario navegar entre ellos y filtrar su búsqueda.
 - Diseñar un reproductor de música flotante que permita al usuario reproducir el single de la página que se está visitando. Manteniendo su estado y persistiendo entre páginas.
 - Asegurarse de que el diseño es responsivo y funcional en distintos dispositivos haciendo uso de Bootstrap

- **Configurar y optimizar** el entorno de **Wordpress**, haciendo uso de **XAMPP** para el **servidor local** y tener un **entorno de staging**.

2.4- Requisitos

2.4.1- Funcionales

- Funcionalidad de **búsqueda** en **tiempo real** haciendo uso de **PHP** (y queries nativas de Wordpress) y **Javascript**.
- Reproductor de música **flotante** usando **Javascript** y la clase de "audio controls".
- Diseño responsivo y dinámico usando Bootstrap.

2.4.2- No Funcionales

- **Optimización** de rendimiento.
- Medidas de **seguridad** consistentes.

2.5- Recursos

2.5.1- Software

- **Frontend:**
 - **Css**, **Bootstrap**, **JavaScript**.
- **Backend:**
 - **PHP** integrado con **WordPress**.
 - Un **IDE** como **Visual Studio Code**.
 - **Git** para el **Control de Versiones**.
- **Database:**
 - **MySQL** manejada por **WordPress**.
- **Plugins:**
 - **Advanced Custom Fields (ACF)**.

2.5.2- Hardware

- Entorno de **desarrollo** para arrancar **XAMPP** y probar la **aplicación**
- Un **servidor** para mantener el **entorno de Wordpress**.

3- Diseño de la solución software

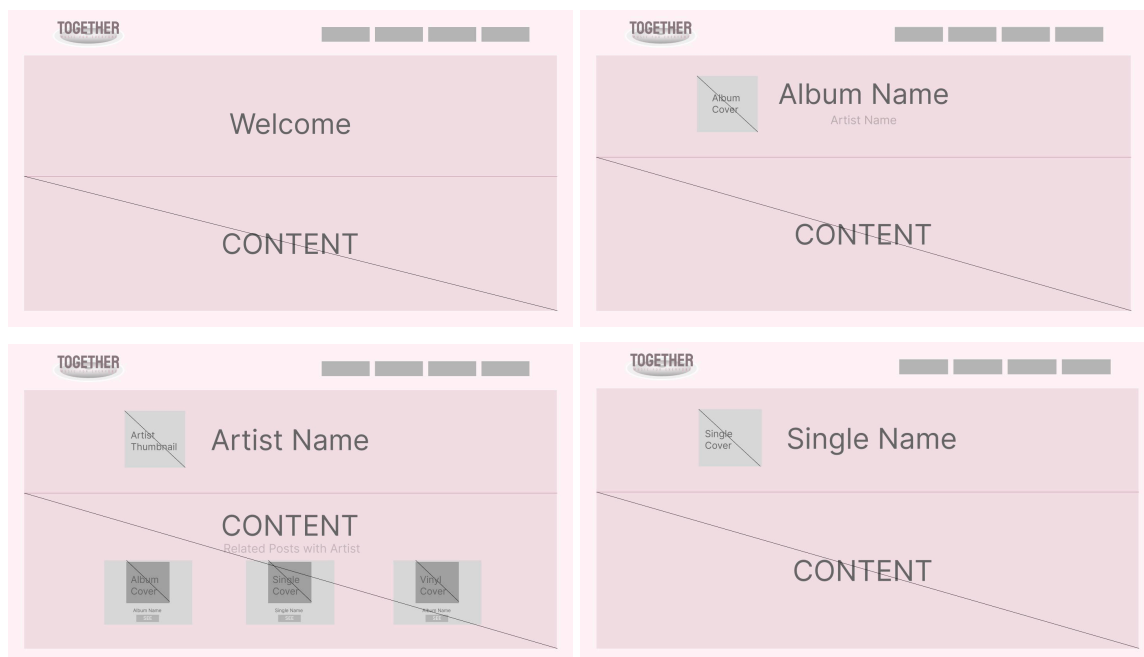
3.1- Modelados

3.1.1- Casos de Uso

- El **usuario interactúa** de distintas formas con la página, se permite a usuarios con un **rol específico** crear tipos de **publicaciones** permitidas (**singles**, y **álbumes**). También se permite escuchar música mediante un **reproductor** . Y realizar **donaciones** a artistas mediante una **integración** de una **API de PayPal**.

3.2- Prototipado Gráfico

3.2.1- [Escritorio]



3.3- Base de Datos

- La base de datos es **MySQL** y está gestionada por **Wordpress**, que se encargará de crear la estructura de datos y almacenarlos en unas tablas predeterminadas (**wp_links**, **wp_options**, **wp_postmeta**, **wp_posts**, **wp_termmeta**, **wp_terms**, **wp_term_relationships**, **wp_term_taxonomy**, **wp_usermeta** y **wp_users**).
- Mediante **phpMyAdmin**, podremos visualizar las tablas, **realizar consultas**, y cualquier caso que surja en relación a la **base de datos**.

4- Implementación

4.1- Codificación

4.1.1- Backend

- La **arquitectura** del **backend** está compuesta por **Php**, funciones específicas del entorno de **Wordpress**, **hooks** y **acciones** en *functions.php*. **Javascript** entre elementos **<script>** en **php** para agregar **efectos** en distintos **elementos** de la **página**. Creación de un nuevo tipo de **rol** llamado **Artist** que permitirá a un usuario registrado **cambiar** su **rol** tras rellenar un **formulario**, y la capacidad de **crear posts** de tipo **Single** y **Álbum**. Si el **rol** del **usuario** es **artista**, se **mostrará** en los **archive** de **Single** y **Álbum** un botón de **Create Single** (o **álbum**), que mostrará un **formulario** que procederá a la **creación** de la **publicación** (si se cumplen todas las **validaciones**). (Si no hay link de PayPal asociado a la publicación del artista no se mostrará el cuadro de donación).

4.1.2- Frontend

- Para el **frontend** se ha usado **Bootstrap** para la estructura de elementos en la página, para asegurar el **diseño responsivo**, y también para dar un **aspecto pulido, cohesivo y organizado**. Algunos elementos más generales se han **estilizado** usando **CSS** para asegurar que se mantienen **persistentes**. También se ha usado Javascript para crear e integrar el reproductor de música flotante que recoge la pista en el post (si la tiene), con persistencia en Localstorage para no perder el progreso ni el estado del mismo a medida que se navega.
- Haciendo uso de Advanced Custom Fields, logramos hacer relaciones entre tipos de posts distintos, y enlace los singles o álbumes a un artista específico. (Se asigna por defecto el creador del post del)

4.2- [Pruebas]

- Se ha comprobado que la **API Paypal for Developers** funciona correctamente , usando el **Sandbox** y **carteras virtuales**, consiguiendo realizar **transacciones** de **PayPal** mediante una **ventana emergente**. (Así nos ahorramos **gestionar datos privados** de compra).
- En el **formulario** de **creación** de **artista**, se **cambia** el **rol** del **usuario** de **Subscriber** a **Artist**, permitiendo así la **creación** de **Singles** y **Álbumes**, y también se **crea** un post de **Artista asociado** al **usuario**.

- Si el **usuario actual** tiene el rol de **Artist**, podrá proceder a **crear un post** de tipo **Single**, en **Archive Singles** mediante un formulario.
- Si el **usuario actual** tiene el rol de **Artist**, también podrá proceder a **crear un post** de tipo **Álbum**, en **Archive Albums** mediante un formulario.

5- Documentación

5.1- Empaquetado y distribución

- Se comprueba que la versión del tema y de Wordpress sean la misma.
- Se crea un fichero .zip de la carpeta del tema de Wordpress (en este caso *twentytwentyfive-child*).
- Se puede **distribuir compartiendo** directamente el **fichero .zip** al **cliente**.
- Y también se puede publicar en **marketplaces** como **ThemeForest**, **Creative Market**, **GitHub**, etc.

5.2- Instalación

1. Desde el panel de Wordpress
 - Ir a **Apariencia > Temas**
 - Hacer clic sobre **Añadir Nuevo > Subir Tema**
 - Seleccionar el archivo ZIP y hacer clic en **Instalar Ahora**.
 - Activar el tema.
2. Instalación manual (**FTP**)
 - Descomprimir el archivo **ZIP**
 - Subir la carpeta a **wp-content/themes/** usando **FTP/SFTP**.
 - Ir a **Apariencia > Temas** en **WordPress** y activar.

6- Conclusiones

- Este **proyecto** ha logrado **desarrollar una plataforma web** en **WordPress** enfocada en el **apoyo a artistas independientes**, brindándoles un **espacio** para **compartir su música** y recibir **donaciones** de los **usuarios**. Se han implementado funcionalidades clave como la **gestión de roles personalizados**, permitiendo a los **artistas crear y administrar sus publicaciones de singles y álbumes**.
- La **integración y creación de un reproductor de música flotante** con **persistencia de estado** mejora la **experiencia del usuario**, mientras que la **implementación de la API de PayPal** facilita las **donaciones sin comprometer la seguridad de los datos**. El uso de tecnologías como **PHP**, **JavaScript** y **Bootstrap** ha **garantizado un diseño responsivo** y una **navegación fluida** en **distintos dispositivos**.

- En conclusión, el proyecto no solo cumple con los **objetivos propuestos**, sino que **sienta las bases** para **futuras mejoras**, como la **implementación** de **suscripciones** o **algoritmos de recomendación** para **potenciar la visibilidad** de los **artistas**.

7- Bibliografía

- Documentación Oficial de WordPress – <https://developer.wordpress.org>
- Bootstrap Documentation – <https://getbootstrap.com/docs/>
- PHP Manual – <https://www.php.net/manual/es/>
- JavaScript MDN Web Docs – <https://developer.mozilla.org/es/docs/Web/JavaScript>
- PayPal for Developers API – <https://developer.paypal.com/docs/api/>
- phpMyAdmin Documentation – <https://www.phpmyadmin.net/docs/>
- Advanced Custom Fields Plugin – <https://www.advancedcustomfields.com/resources/>