**INSTALAR YARN**

npm install --global yarn

**CREAR PROYECTO REACT**

<https://github.com/tupy77/musicfy/commit/80fcac9e51d940f830e8698d951af91daddd8d82>

npx create-react-app nombre\_app

Ahora poniendo yarn start se inicia la aplicación en el puerto 3000

**INCORPORAR ELECTRON AL PROYECTO**

<https://github.com/tupy77/musicfy/commit/10c93bdd4ab37cc296f4734cca6504f9301d362d>

<https://github.com/tupy77/musicfy/commit/560d17790966787aace298569c1257d90edb8b43>

<https://github.com/tupy77/musicfy/commit/d93018f99792ff303d76d9e25fb3f55ddf5d8d09>

**Concurrently** – Para ejecutar mas de un proceso en un solo script de npm

yarn add concurrently -D

el -D es para que se instale en devDependencies

**Electron**

yarn add electron -D

**Wait-on**

Para el tema de cross-platform

yarn add wait-on -D

**Electron-is-dev**

yarn add electron-is-dev

Para una integracion y poder funcionar añadir siguientes cambios:

Crear **.env**

BROWSER=none

Crear **electron.js** dentro de */public*

const *electron* = *require*("electron");

const *app* = *electron.app*;

const *BrowserWindow* = *electron.BrowserWindow*;

const *path* = *require*("path");

const *isDev* = *require*("electron-is-dev");

letmainWindow;

function *createWindow*() {

  mainWindow = new *BrowserWindow*({

    width: 1500,

    height: 1000,

    title: "Musicfy",

    // *titleBarStyle: "hiddenInset",*

    // *resizable: false,*

  });

*mainWindow.loadURL*(

    isDev

      ? "http://localhost:3000"

      : `file://${*path.join*(\_\_dirname, "../build/index.html")}`

  );

  if (isDev) *mainWindow.webContents.openDevTools*();

*mainWindow.on*("closed", () => (mainWindow = null));

}

*app.on*("ready", createWindow);

*app.on*("window-all-closed", () => {

  if (*process.*platform !== "darwin") {

*app.quit*();

  }

});

*app.on*("activate", () => {

  if (mainWindow === null) {

*createWindow*();

  }

});

Modificar **package.json** para introducir el script

"electron-dev": "concurrently \"react-scripts start\" \"electron .\""

A partir de ahora poniendo yarn electron-dev se abrirá nuestra aplicación en la ventana de electron

**ELIMINAR EL WARNING QUE SALE EN LA CONSOLA DEL INSPECTOR DE ELEMENTOS**

<https://github.com/tupy77/musicfy/commit/479cb06ef16201255d0559689c49c092cc71c1fd>

Modificar el **index.html** agregando la siguiente línea

<meta http-equiv="Content-Security-Policy" content="script-src 'self';" />

**INSTALAR SASS Y SISTEMA DE VARIABLES**

<https://github.com/tupy77/musicfy/commit/f898b4751e98f86e2dd9d432117ac02b5cc5956c>

<https://github.com/tupy77/musicfy/commit/9805aa73499a84af80d3bdccd2e50479cb16e92d>

yarn add sass

creamos carpeta *src/scss* con **index.scss color.scss global.scss**

**index.scss**

@*import* "./color.scss";

**color.scss**

$primary-color: #1db954;

$primarycolor-hover: #1ed75f;

$font-light: #fff;

$font-grey: #9e9e9e;

$font-dark: #000;

$border-primary: #282828;

$background-primary: #121212;

$background-seconday: #212121;

$background-dark: #000;

$background-light: #fff;

$danger: #b69494;

**global.scss**

@*import* "./index.scss";

\*,

a,

p {

    color: $font-light;

}

Por ultimo modificamos el **index.js** para que empiece a usarlo

*import* "./scss/global.scss";

**INSTALAR SEMANTIC UI REACT**

<https://github.com/tupy77/musicfy/commit/e0ed932dcc1ad9d5261c1b580d439db2d6dc324b>

yarn add semantic-ui-react semantic-ui-css

importarlo en el **index.js antes del global.scss**

*import* "semantic-ui-css/semantic.min.css";

*import* "./scss/global.scss";

Si fallara, en vez de manera local, hacer la importacion mediante CDN en el **index.html**

<link

async

rel="stylesheet"

href="https://cdn.jsdelivr.net/npm/semantic-ui@2/dist/semantic.min.css"

/>

**USO DE SEMANTIC-UI**

Por ejemplo en el **App.js** importamos los elementos que queramos:

*import* { Button } *from* 'semantic-ui-react'

Luego metemos los botones

      <Button *primary*>Primary</Button>

      <Button *secondary*>Secondary</Button>

Saldrá en la consola un **warning**. Para quitarlo en el **index.js** eliminar el **StrictMode**:

*root.render*(

  <React.StrictMode>

    <App />

  </React.StrictMode>

);

Toda la info y doc sobre semantic esta en:

<https://react.semantic-ui.com/>

**CREAR PROYECTO FIREBASE**

<https://firebase.google.com/?hl=es>

La documentación de firebase para nuestra aplicación es la que esta en “Comienza a usar web”:

<https://firebase.google.com/docs/web/setup?hl=es-419>

Sobre todo vamos a usar en Compilacion Authentication, Firestore y Almacenamiento

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

En la pagina viene en formato namespace (anterior a la versión 8) y la que vamos a estar usando que es la modular:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Dicho esto y habiendonos dado de alta en la pagina, arriba a la derecha al lado de tu avatar saldrá “Ir a la consola” y allí “Agregar proyecto”

Escribimos el nombre del proyecto y continuar y quitamos lo del Google Analytics. Continuams y esperamos a que se cree el proyecto. Cuando termine continuamos y ya tendremos acceso al panel de nuestro proyecto

**AÑADIR FIREBASE A NUESTRA APP**

<https://github.com/tupy77/musicfy/commit/37116587c3a77ab7a7fe3d24e1ae7bc8b5b8beb0>

Le damos a configuración del proyecto:

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Al fondo de la pagina que saldrá pinchamos a app con tecnología web:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Le damos el nombre a la app y le damos a registrar app

Después de esto nos saldrá que tenemos que instalar firebase en nuestro proyecto:

yarn add firebase

Si quieres elegir la versión:

yarn add [firebase@9.8.1](mailto:firebase@9.8.1)

En la pagina de firebase pone que se inicializaría escribiendo esto:

// *Import the functions you need from the SDKs you need*

*import* { initializeApp } *from* "firebase/app";

// *TODO: Add SDKs for Firebase products that you want to use*

// *https://firebase.google.com/docs/web/setup#available-libraries*

// *Your web app's Firebase configuration*

const *firebaseConfig* ={

*apiKey*:"AIzaSyAUHfrVARP9Aom7y1L1G5XPMJMl8B9Gj9Q",

*authDomain*:"musicfy-v1-a394c.firebaseapp.com",

*projectId*:"musicfy-v1-a394c",

*storageBucket*:"musicfy-v1-a394c.appspot.com",

*messagingSenderId*:"252492090801",

*appId*:"1:252492090801:web:ae732f080be833061dc190"

};

// *Initialize Firebase*

const *app* = *initializeApp*(firebaseConfig);

Pero nosotros, para tener ordenado todo creamos carpeta dentro de **src/utils** y ficheros **index.js** y **firebase.js** y en **firebase.js** modificaremos un poco este código:

**firebase.js**

*import* { initializeApp } *from* "firebase/app";

const *firebaseConfig* ={

*apiKey*:"AIzaSyAUHfrVARP9Aom7y1L1G5XPMJMl8B9Gj9Q",

*authDomain*:"musicfy-v1-a394c.firebaseapp.com",

*projectId*:"musicfy-v1-a394c",

*storageBucket*:"musicfy-v1-a394c.appspot.com",

*messagingSenderId*:"252492090801",

*appId*:"1:252492090801:web:ae732f080be833061dc190",

};

*export* const *initFirebase* = *initializeApp*(firebaseConfig);

en **src/utils/index.js**

*export* \* *from* "./firebase";

Haciendolo de esta manera al importar la carpeta utils se importa en automático todo lo que esta en el **index.js**. Por ejemplo si en el **index.js de src** ponemos:

*import* {initFirebase} *from* "./utils";

**COMPROBAR QUE FUNCIONA**

En **App.js**

*import* React *from* "react";

*import* { Button } *from* "semantic-ui-react";

*import* { getAuth } *from* "firebase/auth";

*export* *default* function *App*() {

  const *auth* = *getAuth*();

*console.log*(auth);

…

En la aplicación en la consola debería salir

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**SEPARAR ZONA USUARIOS LOGEADOS Y NO LOGEADOS**

<https://github.com/tupy77/musicfy/commit/2f27cd8f7f9baef3e5c18467f6b7fbef0ae692d3>

**ZONA LOGEADA**

<https://github.com/tupy77/musicfy/commit/8e62b66580a8e801f4c3c410f0583193830ea8bc>

**NAVEGACION USUARIO LOGEADO**

<https://github.com/tupy77/musicfy/commit/c883bdaf309273b2c378332ff14a204663895a12>

Instalar **React-router-dom**

yarn add react-router-dom@6

modificamos archive **LoggedNavigation.js**:

*import* React *from* "react";

*import* { BrowserRouter, Routes, Route } *from* "react-router-dom";

*import* { Album, Albums, Artist, Artists, Home, Profile } *from* "../pages";

*export* function *LoggedNavigation*() {

*return* (

    <BrowserRouter>

      <Routes>

        <Route *path*="/" *element*={<Home />} />

        <Route *path*="/artists" *element*={<Artists />} />

        <Route *path*="/artists/:id" *element*={<Artist />} />

        <Route *path*="/albums" *element*={<Albums />} />

        <Route *path*="/albums/:id" *element*={<Album />} />

        <Route *path*="/profile" *element*={<Profile />} />

      </Routes>

    </BrowserRouter>

  );

}

Si queremos comprobar que funciona lo que hacemos es meter como si hubiera un usuario en **App.js** y ejecutamos la aplicación. En ese caso iría al / que es el Home

App.js

*onAuthStateChanged*(auth, (currentUser) => {

*setUser*({ name: "Teste" });

  });

Y al recargar ya saldría la pagina del home

**NAVEGACION USUARIO NO LOGEADO**

<https://github.com/tupy77/musicfy/commit/34c82dc271e4fb9d95131f96f8b072f4cc7a3091>

Esta navegación la vamos a construir nosotros (Sin react-router-dom) ya que esta no va a tener paginas, es la misma pagina que va a ir cambiando de formulario. Va a estar controlado por un estado y una función

**INTRODUCIENDO ASSETS**

<https://github.com/tupy77/musicfy/commit/96983b5346f3531d2dc69bd1c3c92cea1a51f6d4>

**CONTENEDOR AUTH**

<https://github.com/tupy77/musicfy/commit/d6ef90acac84f52a4a7525a1102553a585c4bd7f>

**OPCIONES DE REGISTRO Y LOGIN**

<https://github.com/tupy77/musicfy/commit/8328d53c6938523255691b7df8987ee8309c9546>

**FORMULARIO DE REGISTRO**

<https://github.com/tupy77/musicfy/commit/98b0c5e028761047c122b0fe96d5d74f52fd9346>

**ESTILOS PARA ERROR EN INPUTS Y BOTON PRIMARY DEL FORM**

<https://github.com/tupy77/musicfy/commit/aab61d074d7596e2d3979046c9de7228875450ef>

**FORMIK, CONTROLAR DATOS DEL FORMULARIO**

<https://github.com/tupy77/musicfy/commit/ece8e6b0c3a2c50108e3eb7a171cf2d1ca78e22e>

yarn add formik

**YUP, VALIDACION DE FORMULARIO**

<https://github.com/tupy77/musicfy/commit/e0e2cf953dc5c9acf207cf0dbd8fd516da720830>

yarn add yup

**REGISTRANDO USUARIO EN BD**

<https://github.com/tupy77/musicfy/commit/4d2d4498ea6cfbb10cb92a33bac25f8c9946c8b0>

Configuramos método de acceso en nuestro firebase

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

En vez de trabajar dentro del mismo archivo integrando firebase, lo vamos a hacer separados para desacoplarlos por si el día de mañana queremos trabajar con otro sistema (AWS, otra BD…).

Creamos carpeta src/api y archivo auth.js

En este caso no tendría que modicar el código de la app sino solamente la clase escrita en este archivo con los nuevos datos

**MOSTRAR/OCULTAR CONTRASEÑA**

<https://github.com/tupy77/musicfy/commit/f0bdc9aff70c9ad16a56d1687018e8d4174de5b0>

**CERRAR SESIÓN**

<https://github.com/tupy77/musicfy/commit/b1cf0f32bfc1bd00b7ddcd6ade459f2f1ede2944>

**FORMULARIO LOGIN**

Estilos:

<https://github.com/tupy77/musicfy/commit/c05089efeb3ea57fb744eb94bfc49d428e9965ee>

Control y validación:

<https://github.com/tupy77/musicfy/commit/95ecfbe7927d5330672d8563a6cf05e26dda7070>

Logear usuario:

<https://github.com/tupy77/musicfy/commit/f56e406cdab7cb158c436ec4e33fe3d0c90acab3>

ESTRUCTURA LAYOUT

<https://github.com/tupy77/musicfy/commit/675a90988d98f2013afb5726511ff0cb088eafcd>

LEFT MENU

<https://github.com/tupy77/musicfy/commit/da828b5254a960622af8bccbf03e3d0246c5bc15>

TOP BAR

Estructura y Estilos

<https://github.com/tupy77/musicfy/commit/32022c7cbc03acb4fa1b6c0dc8bddb2abfa6dc4e>

Funcionalidad:

<https://github.com/tupy77/musicfy/commit/33491901068c9d9325f6282076007c1b38dd37aa>

goBack

logout

controlador user para profile

mostrar usuario y avatar

FOOTER

Estructura y estilos

<https://github.com/tupy77/musicfy/commit/dde3cc5e745732b2e479c6fc492e561297330848>

REPRODUCTOR FOOTER

Estructura

<https://github.com/tupy77/musicfy/commit/24da471ef187f3cb2d6a5ef601aa7c84a4e90b7b>

AJUSTES DE USUARIO

Estructura y estilos

<https://github.com/tupy77/musicfy/commit/6337755b8e2a7e10d4ec79f4d3af649240b617d1>

COMPONENTE AVATAR

Estructura y estilo

<https://github.com/tupy77/musicfy/commit/f15588b546fb8d9f5b7acbae94c989371c264bf8>

Funcionalidad y carga de foto

Importar react dropzone

yarn add react-dropzone

Apretar icono y que salga el explorador de archivos

<https://github.com/tupy77/musicfy/commit/ab4de6842c55ee4829fb644755ac299f0e6f4bb7>

Poder arrastrar o elegir una foto y que se vea en el avatar sin llegar a subirse a la base de datos

<https://github.com/tupy77/musicfy/commit/73d504d20836027e2467baa86d57de41e5e3f866>

SUBIR AVATAR A FIREBASE

Entramos en nuestro proyecto de firebase/compilación/storage para configurarlo

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Le damos a “Comenzar” y decimos (en un principio) que lo queremos en modo producción

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Elegimos la ubicación del servidor (no se podrá cambiar)

Se cambian las reglas

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Y se pone que todo el mundo puede escribir y leer en “true”.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Después le damos a publicar

Ya lo demás es el código

<https://github.com/tupy77/musicfy/commit/c80dd611895b87da3613426df4f23c8fca5472ca>

USAR AVATAR DE FIREBASE

<https://github.com/tupy77/musicfy/commit/5c3274d953fabc0364eb3fadf577c7a20cea0ae6>

PROFILE CREANDO EL MODAL ACTUALIZAR DATOS

<https://github.com/tupy77/musicfy/commit/3903f2315da14fb30c3dd3e5602a52a466b8919e>

PROFILE MODAL CONTROLAR CONTENIDO

<https://github.com/tupy77/musicfy/commit/b624127de41fd578ec4b7f36b824e9d3c795f680>

FORMULARIO DISPLAYNAME

Estructura

<https://github.com/tupy77/musicfy/commit/8a595cc081dae3216547f7627b5406d71865c060>

Control y validación de datos

<https://github.com/tupy77/musicfy/commit/2254091eb65697a30505ca4830eb63edb5b3343b>

Actualizando datos

<https://github.com/tupy77/musicfy/commit/b5f522ce221c9b758a41149b8a00f509169ee1e4>

FORM EMAILUPDATE

<https://github.com/tupy77/musicfy/commit/cd990466d11c7a8aeafa38ddc8d64a8e40cc7f4d>

<https://github.com/tupy77/musicfy/commit/084000467044f2d70e6e896e34b17b72538eb2e5>

FORM PASSWORD UPDATE

<https://github.com/tupy77/musicfy/commit/e8c3ecce5382bda3b78310b5c53cb140c09bc71b>

MENSAJES DE ERROR AL INTRODUCIR DATOS INCORRECTOS

<https://github.com/tupy77/musicfy/commit/8eab12ed40df90cc7d400d88c8b5f4e24a53ba28>

MODAL PARA ABRIR FORMULARIOS DE CREACION (ARTISTAS, CANCIONES…)

<https://github.com/tupy77/musicfy/commit/5236c04cd0cc6e87a6c29d8aa0d74f9f1c2a5096>

FORM CREAR ARTISTA

Estructura

<https://github.com/tupy77/musicfy/commit/237c0c16008c0032f267692305f84aced3f71c40>

Control y validación:

Como hay un div dentro del formulario hay que instalar la dependencia classnames para poder asignar clases con condicionales

yarn add classnames

<https://github.com/tupy77/musicfy/commit/fceee344f7e1375673729e55cb8109a94376d077>

Control imagen:

<https://github.com/tupy77/musicfy/commit/de719e16644725c3dfcb28b76f0100a40b77840b>

Crear artista

<https://github.com/tupy77/musicfy/commit/fb2875033f48801cb968d83863cbbb37e55163ff>

Activamos en Firebase el Firestore Database. Esto es Base de datos y lo otro era un almacenamiento.

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Le damos a crear base de datos y lo ponemos en modo de producción. Elegimos el servidor y creamos. Después modificamos las reglas para poner a “true” que se pueda leer y escribir

Instalamos uuid

yarn add uuid

OBTENER TODOS LOS ARTISTAS

<https://github.com/tupy77/musicfy/commit/05bc87b8aa01b637afd7006311179c68c71ef7b9>

Instalar lodash para usar map. Es un paquete que trae funciones básicas de javascript pero potenciadas

yarn add lodash

MOSTRAR TODOS LOS ARTISTAS

<https://github.com/tupy77/musicfy/commit/4049edaba47f95668ade97b2227f4edafd1962d7>

OBTENER DATOS DE UN ARTISTA

<https://github.com/tupy77/musicfy/commit/12d5c17a39a6c232206990fcdcd52d348f30e50e>

MOSTRAR DATOS DE UN ARTISTA

<https://github.com/tupy77/musicfy/commit/47a77d0880771b094d599ec64317e0ae491d22a0>

FORM CREAR ALBUNES

Estructura:

<https://github.com/tupy77/musicfy/commit/a93ff9fd1abf1b50ea19880eb306840b6530ffb2>

Validaciones:

<https://github.com/tupy77/musicfy/commit/57d78064671d8c281388d76ee83f5ba15a949a93>

Control imagen

<https://github.com/tupy77/musicfy/commit/32ba47fb7aef89c237bfa35e3729d9e401fdbf2f>

Cargar artistas en DROPDOWN

<https://github.com/tupy77/musicfy/commit/c21b79358ab0892f6f3a167f4e912ebbf8cd4f42>

CREAR ALBUM

<https://github.com/tupy77/musicfy/commit/0a1c68227c8f75f293d0fb4a88d7d207c2403d7c>

OBTENER ALBUMS

<https://github.com/tupy77/musicfy/commit/3116f4c8a4bd30f75d9c27075835ac5c8082ee02>

MOSTRAR ALBUMS

<https://github.com/tupy77/musicfy/commit/237911c442b3930d2197fbd580eeff23414c678b>

OBTENER DATOS DE UN ALBUM

<https://github.com/tupy77/musicfy/commit/8a7de2f1768256c1b10f36b9826cd25a27a60a5d>

MOSTRAR DATOS DE UN ALBUM

<https://github.com/tupy77/musicfy/commit/08ea23eb2640ee1bcfbf28daef6810c36cb4966a>

DATOS ARTISTA ALBUM INVITADO

<https://github.com/tupy77/musicfy/commit/45e47f688c55e12fed56853dcd0a8cdb66a23645>

OBTENER ALBUMS DEL ARTISTA VISITADO (ir a su perfil)

<https://github.com/tupy77/musicfy/commit/4ebe8b32be6e79f8e87f6789b4020bc882f0b183>

RENDERIZANDO SLIDER DE ALBUMES

<https://github.com/tupy77/musicfy/commit/455ab85f8b26c0903312d1b8ed03e22f298552df>

Instalar react-slick y slick-carousel

yarn add react-slick slick-carousel

Hay que repararlo, no se porque me salen dos veces la informacion

FORMULARIO SUBIR CANCION

<https://github.com/tupy77/musicfy/commit/d1548fffd47ae31782a1d58a7e9c8e7aaa0eeeb5>

CONTROL Y VALIDACION

<https://github.com/tupy77/musicfy/commit/f5857d74bc5bb2252f56ad09c8a16e0784b4914a>

CONTROLANDO FICHERO DE MUSICA

<https://github.com/tupy77/musicfy/commit/6d433cb5a38ad94095a79c8d2b20181faa3806cb>

CONTROLANDO ALBUMES

<https://github.com/tupy77/musicfy/commit/461a594abebfe04e9cd810eec9acd147564c0734>

SUBIENDO CANCION

<https://github.com/tupy77/musicfy/commit/5409a3259735b5336882079fd0d1158bda30b19c>

OBTENER CANCIONES DEL ALBUM VISITADO

<https://github.com/tupy77/musicfy/commit/5b329bca7f731980858b4f27909d0a1aba59ce41>

MOSTRAR CANCIONES ALBUM VISITADO

<https://github.com/tupy77/musicfy/commit/b6b9ad2a6a1958b55492ac1838090f50b60dea76>

OBTENIENDO CANCIONES DE UN ARTISTA

<https://github.com/tupy77/musicfy/commit/a61fb81039991cebd0246971c93bc177888e794c>

FIX SLIDER

<https://github.com/tupy77/musicfy/commit/ba48d270090c82601dfd2207d11ba5e438c5ad9b>

MOSTRAR CANCIONES DE UN ARTISTA

<https://github.com/tupy77/musicfy/commit/276542400eb27b404b062b322fe6f7a6a89df489>

ADAPTAR SLIDER

<https://github.com/tupy77/musicfy/commit/53ca6b5c9489ba8c4160e4d3e57a9b2709edfeb1>

**HOMESCREEN**

AÑADIENDO BANNER PRINCIPAL y OBTENER Y MOSTRAR ARTISTAS, ALBUMES Y CANCIONES

<https://github.com/tupy77/musicfy/commit/4a8966b2c121caede043a8b0c445e1497cdbec9f>

FIX IMÁGENES

<https://github.com/tupy77/musicfy/commit/76b0c522a80dfa76c9b83b534877e2ca8b35ff8f>

FIX SHOW SONG'S PICTURE

<https://github.com/tupy77/musicfy/commit/d27687511ee8964385f4577d56eb9cfb7c2c79c8>

**REPRODUCTOR DE MUSICA**

CREAR EL CONTEXTO DEL REPRODUCTOR

<https://github.com/tupy77/musicfy/commit/fe2e640cada37b3b43e08fa5cb9a3f0a2ef35076>

CONTROLAR CANCION QUE SE REPRODUCIRA

<https://github.com/tupy77/musicfy/commit/9918ae7774dd6f1e33372c5b7d87caf4048a1032>

CONTROL VOLUME

<https://github.com/tupy77/musicfy/commit/9faf977f5baafd8905e4a5c2daa03feae3e5666b>

IMPLEMENTAR REPRODUCTOR

<https://github.com/tupy77/musicfy/commit/62fe0228b4a308419f875e84a5b639d69b908d1c>

Instalar reactPlayer

yarn add react-player

IMPLEMENTAR CONTROLES REPRODUCTOR

<https://github.com/tupy77/musicfy/commit/026e9e46153279dd6902d198704faf494afc9547>

BARRA PROGRESO REPRODUCCION & FIX others

<https://github.com/tupy77/musicfy/commit/44dbf4fedb1b12e5ba572a0be69df6f1798ae72f>

**ELECTRON**

**CONFIGURAR ELECTRON**

CONFIGURAR ANCHO Y ALTO

<https://github.com/tupy77/musicfy/commit/59e20a45b7a7ed11f99d57cf9d7f6f778f0e6d5d>

BLOQUEAR O NO BLOQUEAR EL RESIZABLE

<https://github.com/tupy77/musicfy/commit/abd0803e539b93e0e92909ee8057a63683fe7468>

APP MODO KIOSKO

Que ocupe toda la pantalla y que no haya barra ni nada de eso. Se quitaría con Ctrl+w

<https://github.com/tupy77/musicfy/commit/0dc68f131bc962f0fac2222fab94c321b0b4692f>

CAMBIANDO TITULO DE LA APP

<https://github.com/tupy77/musicfy/commit/a24499c5586a9ab794353e4589f4d8ae5776c759>

AÑADIENDO BOTONES MAX, MIN, CERRAR

<https://github.com/tupy77/musicfy/commit/505f9d3896a2ac5bd1222079a7ff13bc9e0ec31e>

CAMBIAR ICONO

<https://github.com/tupy77/musicfy/commit/f7bb3b5cea91bcbc8a4bbe5103de6c6f0933f57a>

**COMPILAR LA APP**

CONFIGURACION PARA COMPILAR

Cambiar el package.json y comentar una línea del index.html

**#EN PACKAGE.JSON**

"main": "public/electron.js"

"author": "Agustín Navarro Galdon"

"homepage": "./"

**#DENTRO DE SCRIPTS**

"electron-pack": "electron-builder -c.extraMetadata.main=build/electron.js"

"preelectron-pack": "yarn build"

**#DEBAJO DE DEVDEPENDENCIES**

"build": {

"appId": "com.agustinnavarrogaldon.musicfy",

"files": [

"build/\*\*/\*",

"node\_modules/\*\*/\*"

],

"directories": {

"buildResources": "assets"

}

}

CREAR .DMG PARA MACOS

yarn electron-pack –mac

CREANDO EL .EXE PARA WINDOWS

El BrowserRouter en ./routes/LoggedNavigation.js da problemas. CAMBIAR POR HashRouter

CON ELECTRON BUILDER

yarn add electron-builder -D

introducir la siguiente linea en los scripts del package.json

    "electron-pack": "electron-builder -c.extraMetadata.main=build/electron.js",

yarn electron-pack –win

Si falla al compilar, instalar Tambien el electron-packager

Se crearan dos carpetas. La build y la dist

En la build esta el html, css y js con la que la aplicación iría correctamente a través del navegador

En la dist habría un instalador y además una carpeta en la que ya seria app instalada

CON ELECTRON PACKAGER (NO FUNCIONA ASI, NO COGE EL INDEX.HTML)

Agregar el siguiente script:

yarn add electron-packager

        "package-win": "electron-packager . Musicfy --overwrite --asar --platform=win32 --arch=x64 --icon=logo-green.ico --prune=true --out=release-builds --version-string.CompanyName=CE --version-string.FileDescription=CE --version-string.ProductName=\"Musicfy by Franci\""

yarn package-win

Se crear una carpeta llamada reléase-builds. En ella estará la aplicación